EXHIBIT D

PLR 4-3(b) – Microsoft's Listing of Intrinsic and Extrinsic Evidence

Set forth below are references to the "instrinsic" and "extrinsic" evidence on which Microsoft may rely to support its claim construction for the 30 designated "Mini-Markman" terms and phrases. Each claim phrase incorporates the intrinsic and extrinsic support of the individual terms within it.

For ease of reference, the full titles of various intrinsic and extrinsic evidence sources are abbreviated. A key to the abbreviations is contained in Appendix 1, located at the last page of this Exhibit.

	Claim Term/Phrase	Evidence Supporting MS Construction
1.	aspect 683.2 861.58 900.155 912.8	**Intrinsic:** 1. "For each site, the manufacturer generates a site ID 2821 and list of site characteristics 2822." ('193 209:55-57) 2. See also support listed in item #29 ('900:155) **Extrinsic:** 1. Aspect: "The qualification of a descriptor." (IBM)
2.	authentication 193.15	**Intrinsic:** 1. "A certification key pair may be used as part of a 'certification' process for PPEs 650 and VDE electronic appliances 600. This certification process in the preferred embodiment may be used to permit a VDE electronic appliance to present one or more 'certificates' authenticating that it (or its key) can be trusted. As described above, this 'certification' process may be used by one PPE 650 to 'certify' that it is an authentic VDE PPE, it has a certain level of security and capability set (e.g., it is hardware based rather than merely software based), etc." ('193 212:66 - 213:15) 2. "One of the functions SPU 500 may perform is to validate/authenticate VDE objects 300 and other items. Validation/authentication often involves comparing long data strings to determine whether they compare in a predetermined way." ('193 67:56-60) 3. "Sender 4052 may select different ways to identify recipients 4056 based on the confidentiality of the document and the level of security the sender is willing to pay for. In one example, sender 4052 might require the recipient's appliance 600B to require recipient 4056 to prove that he is who he says he is. This secure 'authentication' function might be met by, for example, requiring recipient 4056 to input a password, present digital proof of identity..." ('683 17:20-27) 4. "In order to further assure the authenticity of the communication, a secure communications link may be established using a key exchange technique (e.g., Diffie-Hellman) and encryption of the signal between the stations." ('683 52:56-60) 5. "This 'channel 0' 'open channel' task may then issue a series of requests to secure database manager 566 to obtain the 'blueprint' for constructing one or more component assemblies 690 to be associated with channel 594 (block 1127). In the preferred embodiment, this 'blueprint' may comprise a PERC 808 and/or URT 464. In may be obtained by using the 'Object, User, Right' parameters passed to the 'open channel' routine to 'chain' together object registration table

	Claim Term/Phrase	Evidence Supporting MS Construction
		460 records, user/object table 462 records, URT 464 records, and PERC 808 records. This 'open channel' task may preferably place calls to key and tag manager 558 to validate and correlate the tags associated with these various records to ensure that they are authentic and match. The preferred embodiment process then may write appropriate information to channel header 596 (block 1129)." ('193 112:46-61) **Extrinsic:** 1. Authentication: "1. In computer security, verification of the identity of a user or the user's eligibility to access an object. 2. In computer security, verification that a message has not been altered or corrupted. 3. In computer security, a process used to verify the user of an information system or protected resources. 4. A process that checks the integrity of an entity." (IBM) 2. Authentication: "1. In data security, the act of determining that a message has not been changed since leaving its point of origin. ... 4. In computer security, the act of identifying or verifying the eligibility of a station, originator, or individual to access specific categories of information." (Longley)
3.	budget 193.1	**Intrinsic:** 1. "'Budgets' 308 shown in FIG. 5B are a special type of 'method' 1000 that may specify, among other things, limitations on usage of information content 304, and how usage will be paid for. Budgets 308 can specify, for example, how much of the total information content 304 can be used and/or copied. The methods 310 may prevent use of more than the amount specified by a specific budget." ('193 59:19-25) (See also Fig. 5B) 2. "For example, consider the case of a security budget. One form of a typical budget might limit the user to 10Mb of decrypted data per month." ('193 265:9-11) 3. "An example of the process steps used for the move of a budget record might look something like this: 1) Check the move budget (e.g., to determine the number of moves allowed)" ('193 265:24-27) 4. "BUDGET method 408 may store budget information in a budget UDE..." ('193 182:25-26) 5. "BUDGET method 408 may result in a 'budget remaining' field in a budget UDE being decremented by an amount specified by BILLING method 406." ('193 182:27-30) 6. "In the preferred embodiment, a 'method' 1000 is a collection of basic instructions, and information related to basic instructions, that provides context, data, requirements and/or relationships for use in performing, and/or preparing a perform, basic instructions in relation to the operation of one or more electronic appliances 600." ('193 85:43-48; see also '193 136:20-25) 7. "Budget process 408 limits how much content usage is permitted. For example, budget process 408 may limit the number of times content may be accessed or copied, or it may limit the number of pages or other amount of content that can be used based on, -for example, the number of dollars available in a credit account. Budget process 408 records and reports financial and other transaction information associated with such limits." ('193 58:27-34)

	Claim Term/Phrase	Evidence Supporting MS Construction
		8. "BUDGET method 1510 may next perform a billing operation by adding a billing amount to a budget value (block 1602)." ('193 187:48-50) 9. "The permissions and/or methods (i.e., budgets) carried by the portable appliance 2600 may have been assigned to it in conjunction with an 'encumbering' of another, stationary or other portable VDE electronic appliance 600." ('193 235:39-42) 10. "Fields used for budget (but not for meter): 'Descending use counter ... Start date'" ('193 143:63 - 144:14) 11. "A budget may be specified in dollars, deutsche marks, yen, and/or in any other monetary or content measurement schema and/or organization. The preferred embodiment output of the application, normally has three basic elements. A notation in the distribution portion of secure database 610 for each budget record created, the actual budget records, and a method option record for inclusion in a permissions record." ('193 265:44-51) **Extrinsic:** 1. Budget: "A budget is the control mechanism for a meterable feature. A budget provides an upper limit for the volume of a meterable feature that a user (client) may use. Budgets consist of two values: a ceiling limit on use and an increment value that is added to the associated meter when a meterable event occurs. Budgets may be stand-alone or cascaded. A stand-alone budget only increments the meters for itself, while a cascaded budget can increment many meters from a single meterable event. A budget consists of an identification sextet, a descriptive area that describes the budget (cascade budget tuple and other miscellaneous flags), and a series of budget tuples. Each budget tuple consists of a budget and the increment value. It should be noted that a budget may be specified in meterable events or in dollars, based on the type of meter the budget will be compared against." (VDE ROI Device v1.0a, 2/9/94, IT00008582) 2. Budget Object: "A governed element that defines the consumer's ability to provide payment using a specific payment type." (IT Glossary[1], 1997-1998, ML00012B) 3. Budget Object: "*An InterTrust system object* that defines the consumer's ability to provide payment using a specific payment type." (emphasis added) (IT System Developers Kit, 1997, TD00298C) 4. Budget: "A control mechanism that limits operations on content based on billed amounts that can maintain a budget trail. A budget may be financially based (e.g., a number of dollars available for purchasing content use) or abstract (e.g. a total number of permitted usages)." (IT Glossary, 3/7/95, IT00709617) 5. Budget: "*A fixed quantity of money, time, etc. against which the cost of operation is charged. Budget activities usually also involve reporting." (IT Glossary, 8/21/95, IT0032371)

[1] "IT Glossary" herein is a generic reference to several "glossaries" that have been created by InterTrust and that are further identified by Bates number and/or IT document number.

	Claim Term/Phrase	Evidence Supporting MS Construction
4.	clearinghouse 193.19	**Intrinsic:** 1. "Distribution involves three types of entity. Creators usually are the source of distribution. They typically set the control structure 'context' and can control the rights which are passed into a distribution network. Distributors are users who form a link between object (content) end users and object (content) creators. They can provide a two-way conduit for rights and audit data. Clearinghouses may provide independent financial services, such as credit and/or billing services, and can serve as distributors and/or creators. Through a permissions and budgeting process, these parties collectively can establish fine control over the type and extent of rights usage and/or auditing activities." ('193 267:34-45) 2. "Payment credit or currency may then be automatically communicated in protected (at least in part encrypted) form through telecommunication of a VDE container to an appropriate party such as a clearinghouse, provider of original property content or appliance, or an agent for such provider (other than a clearinghouse)." ('193 36:64 - 37:3) 3. "...if appropriate credit (e.g. an electronic clearinghouse account from a clearinghouse such as VISA or AT&T) is available..." ('193 25:22-24) **Extrinsic:** 4. Clearinghouse: "*A facility that receives reports of content use and in turn reports payments and usage to content creators and distributors." (IT Glossary, 8/21/95, TD00068B, IT00032372)
5.	compares 900.155	**Intrinsic:** 1. "ROS 602 also provides a tagging and sequencing scheme that may be used within the loadable component assemblies 690 to detect tampering by substitution. Each element comprising a component assembly 690 may be loaded into an SPU 500, decrypted using encrypt/decrypt engine 522, and then tested/compared to ensure that the proper element has been loaded. Several independent comparisons may be used to ensure there has been no unauthorized substitution. For example, the public and private copies of the element ID may be compared to ensure that they are the same, thereby preventing gross substitution of elements." ('193 87:41-51) **Extrinsic:** 1. Compare: "1. To examine two items to discover their relative magnitudes, their relative positions in an order or in a sequence, or whether they are identical in given characteristics. 2. To examine two or more items for identity, similarity, equality, relative magnitude, or order in a sequence." (IBM) 2. Comparison: "The process of examining two or more items for identity, similarity, equality, relative magnitude, or for order in sequence." (IBM)
6.	component assembly 912.8, 912.35	**Intrinsic:** 1. "Many such load modules are inherently configurable, aggregatable, portable, and extensible and singularly, or in combination (along with associated data), run as control methods under the VDE transaction operating environment." ('193 25:48-52)

	Claim Term/Phrase	Evidence Supporting MS Construction
		2. "Much of the functionality provided by ROS 602 in the preferred embodiment may be based on 'components' that can be securely, independently deliverable, replaceable and capable of being modified (e.g., under appropriately secure conditions and authorizations). Moreover, the 'components' may themselves be made of independently deliverable elements. ROS 602 may assemble these elements together (using a construct provided by the preferred embodiment called a 'channel') at execution time. For example, a 'load module' for execution by SPU 500 may reference one or more 'method cores,' method parameters and other associated data structures that ROS 602 may collect and assemble together to perform a task such as billing or metering. Different users may have different combinations of elements, and some of the elements may be customizable by users with appropriate authorization." ('193 77:12-27) 3. "As discussed above, ROS 602 in the preferred embodiment is a component-based architecture. ROS VDE functions 604 may be based on segmented, independently loadable executable 'component assemblies' 690. These component assemblies 690 are independently securely deliverable. The component assemblies 690 provided by the preferred embodiment comprise code and data elements that are themselves independently deliverable. Thus, each component assembly 690 provided by the preferred embodiment is comprised of independently securely deliverable elements which may be communicated using VDE secure communication techniques, between VDE secure subsystems. These component assemblies 690 are the basic functional unit provided by ROS 602. The component assemblies 690 are executed to perform operating system or application tasks. Thus, some component assemblies 690 may be considered to be part of the ROS operating system 602, while other component assemblies may be considered to be 'applications' that run under the support of the operating system." ('193 83:11-22) 4. "A complete VDE process to service a 'use event' may typically be constructed as a combination of methods 1000." ('193 181:20-21) 5. "The audit information may be, in part, or whole, in some summary and/or analyzed form further processed at the clearinghouse and/or may be combined with other information to form a, at least in part, derived set of information and inserted into one or more at least in part secure VDE objects to be communicated to said one or more (further) auditor parties." ('193 272:29-36) 6. "Components 690 are preferably designed to be easily separable and individually loadable. ROS 602 assembles these elements together into an executable component assembly 690 prior to loading and executing the component assembly (e.g., in a secure operating environment such as SPE 503 and/or HPE 655)." ('193 83:43-48) 7. "component assemblies 690" ('193 83:23); see also "components 690" ('193 86:51-52) 8. "In the preferred embodiment, ROS 602 assembles component assemblies 690 based on the following types of elements: Permissions Records ('PERC's) 808; Method 'Cores' 1000; Load Modules 1100; Data Elements (e.g., User Data Elements ('UDEs') 1200 and Method Data Elements ('MDEs') 1202); and Other component assemblies 690." ('193 85:21-29)

	Claim Term/Phrase	Evidence Supporting MS Construction
		9. "...creation of component assemblies 690 from independently deliverable modules such as method cores 1000, load modules 1100, and data structures such as UDEs 1200." ('193 170:2-4) 10. "ROS 602 also provides a tagging and sequencing scheme that may be used within the loadable component assemblies 690 to detect tampering by substitution. Each element comprising a component assembly 690 may be loaded into an SPU 500, decrypted using encrypt/decrypt engine 522, and then tested/compared to ensure that the proper element has been loaded. Several independent comparisons may be used to ensure there has been no unauthorized substitution. For example, the public and private copies of the element ID may be compared to ensure that they are the same, thereby preventing gross substitution of elements. In addition, a validation/correlation tag stored under the encrypted layer of the loadable element may be compared to make sure it matches one or more tags provided by a requesting process. This prevents unauthorized use of information. As a third protection, a device assigned tag (e.g., a sequence number) stored under an encryption layer of a loadable element may be checked to make sure it matches a corresponding tag value expected by SPU 500. This prevents substitution of older elements. Validation/correlation tags are typically passed only in secure wrappers to prevent plaintext exposure of this information outside of SPU 500." ('193 87:41-62) 11. "Memory manager 578 and virtual memory manager 580 in the preferred embodiment manage ROM 532 and RAM 534 memory within SPU 500 in the preferred embodiment. Virtual memory manager 580 provides a fully 'virtual' memory system to increase the amount of 'virtual' RAM available in the SPE secure execution space beyond the amount of physical RAM 534a provided by SPU 500. Memory manager 578 manages the memory in the secure execution space, controlling how it is accessed, allocated and deallocated. SPU MMU 540, if present, supports virtual memory manager 580 and memory manager 578 in the preferred embodiment. In some 'minimal' configurations of SPU 500 there may be no virtual memory capability and all memory management functions will be handled by memory manager 578. Memory management can also be used to help enforce the security provided by SPE 503. In some classes of SPUs 500, for example, the kernel memory manager 578 may use hardware memory management unit (MMU) 540 to provide page level protection within the SPU 500. Such a hardware-based memory management system provides an effective mechanism for protecting VDE component assemblies 690 from compromise by 'rogue' load modules." ('193 109:24-45) 12. "The channel 594 and its header 596 comprise a data structure that 'binds' or references elements of one or more component assemblies 690. Thus, the channel 594 is the mechanism in the preferred embodiment that collects together or assembles the elements shown in FIG. 11E into a component assembly 690 that may be used for event processing." ('193 115:65 - 116:4) 13. "It reads the appropriate open control elements from the secure database (or the container, such as, for example, in the case of a traveling object), and 'binds' or 'links' these particular appropriate control elements together in order to control opening of the object for this user." ('193 185:42-46)

	Claim Term/Phrase	Evidence Supporting MS Construction
		14. "Thus, PERC 808 in effect contains a 'list of assembly instructions' or a 'plan' specifying what elements ROS 602 is to assemble together into a component assembly and how the elements are to be connected together. PERC 808 may itself contain data or other elements that are to become part of the component assembly 690." ('193 85:30-39) 15. "The selected method event record 1012, in turn, specifies the appropriate information (e.g., load module(s) 1100, data element UDE(s) and MDE(s) 1200, 1202, and/or PERC(s) 808) used to construct a component assembly 690 for execution in response to the event that has occurred. ..." ('193 138:31-36) 16. "As mentioned above, ROS 602 provides several layers of security to ensure the security of component assemblies 690. One important security layer involves ensuring that certain component assemblies 690 are formed, loaded and executed only in secure execution space such as provided within an SPU 500. Components 690 and/or elements comprising them may be stored on external media encrypted using local SPU 500 generated and/or distributor provided keys." ('193 87:33-40) 17. "ROS 602 also provides a tagging and sequencing scheme that may be used within the loadable component assemblies 690 to detect tampering by substitution." ('193 87:41-43) 18. "ROS 602 generates component assemblies 690 in a secure manner. As shown graphically in FIGS. 11I and 11J, the different elements comprising a component assembly 690 may be 'interlocking' in the sense that they can only go together in ways that are intended by the VDE participants who created the elements and/or specified the component assemblies. ROS 602 includes security protections that can prevent an unauthorized person from modifying elements, and also prevent an unauthorized person from substituting elements." ('193 84:60 - 85:2) 19. "ROS 602 assembles these elements together into an executable component assembly 690 prior to loading and executing the component assembly (e.g., in a secure operating environment such as SPE 503 and/or HPE 655). ROS 602 provides an element identification and referencing mechanism that includes information necessary to automatically assemble elements into a component assembly 690 in a secure manner prior to, and/or during, execution." ('193 83:44-52) 20. "Wherein said processor includes: retrieving means for retrieving at least one component, and at least one record that specifies a component assembly, from said memory devices, checking means coupled to said retrieving means for checking said component and/or said record for validity, and using means coupled to said retrieving means for using said component to form said component assembly in accordance with said record." ('107 Application p. 782 claim 80) 21. "These called-for method(s) and data structure(s) (e.g., load modules 1100, UDEs 1200 and/or MDEs 1202) are each decrypted using encrypt/decrypt manager 556 (if necessary), and are then each validated using key and tag manager 558. Channel manager 562 constructs any necessary 'jump table' references to, in effect, 'link' or 'bind' the elements into a single cohesive executable so the load module(s) can reference data structures and any other load module(s) in the

	Claim Term/Phrase	Evidence Supporting MS Construction
		component assembly. Channel manager 562 may then issue calls to LMEM 568 to load the executable as an active task." ('193 116:25-35) **Extrinsic:** 1. Component: "1. Hardware or software that is part of a functional unit. 2. A functional part of an operating system. 3. A set of modules that performs a major function within a system." (IBM) 2. Component: "In data communications, a device or set of devices, consisting of hardware, along with its firmware, and or software that performs a specific function on a computer communications network. A Component is a part of a larger system, and may itself consist of other components." (Longley) 3. Record: "1. In programming languages, an aggregate that consists of data objects, possibly with different attributes, that usually have identifiers attached to them. In some programming languages, records are call structures. 2. A set of data treated as a unit. 3. A set of one or more related data items grouped for processing." (IBM) 4. Record: "1. In computing, a collection of related data treated as a unit, e.g. details of name, address, age, occupation and department of an employee in a personnel file. 2. In computing, to store signals on a recording medium for later use." (Longley) 5. Record: "1. A collection of related data or words treated as a unit and saved in a position dependent fashion within a file or other such unit. 2. A set of data items, called fields, treated as a unit." (Booth) 6. Secure: "Pertaining to the control of who can use an object and to the extent to which the object can be used by controlling the authority given to the user." (IBM)
7.	contain 683.2 912.8, 912.35	**Intrinsic:** 1. "Container 300y may contain and/or reference rules and control information 300y(1) that specify the manner in which searching and routing information use and any changes may be paid for." ('193 241:36-39) 2. "Each logical object structure 800 may also include a 'private body' 806 containing or referencing a set of methods 1000 (i.e., programs or procedures) that control use and distribution of the object 300." ('193 128:25-28) 3. "Therefore, stationary object structure 850 does not contain a permissions record (PERC) 808; rather, this permissions record is supplied and/or delivered separately (e.g., at a different time, over a different path, and/or by a different party) to the appliance/installation 600." ('193 130:18-22) 4. "The content portion of a logical object may be organized as information contained in, not contained in, or partially contained in one or more objects." ('193 127:8-19) 5. "Container 302 may 'contain' items without those items actually being stored within the container. For example, the container 302 may reference items that are available elsewhere such as in other containers at remote sites. Container 302 may reference items available at different times or only during limited times. Some items may be too large to store within container 302. Items may, for example, be delivered to the user in the form of a 'live feed' of video at a certain

	Claim Term/Phrase	Evidence Supporting MS Construction
		time. Even then, the container 302 'contains' the live feed (by reference) in this example." ('193 58:49-58) 6. "Load modules 1100 may contain or reference other load modules." ('193 86:47-48) 7. "PERC 808(k) defines, among other things, the 'assembly instructions' for component assembly 690(k), and may contain or reference parts of some or all of the components that are to be assembled to create a component assembly." ('193 87:3-6) 8. "Alternatively, traveling object PERCs 808 may contain or reference budget records..." ('193 130:63-64) 9. "Method 'core' 1000' in the preferred embodiment may contain or reference one or more data elements such as MDEs 1202 and UDEs 1200." ('193 136:32-34) 10. "Container 300y may contain and/or reference rules and control information 300y(1) that specify the manner in which searching and routing information use and any changes may be paid for." ('193 241:36-39) 11. "Trusted go-between 4700 registers the contract 4068, and then creates an electronic list of rules based on contract 4068. A partial example rule list is shown in FIG. 130A. Although the FIG. 130A conditions are shown as being written on a clipboard, in the preferred embodiment the" ('683 54:29-37) 12. See also prior art referred to in the relevant InterTrust patent file histories, e.g. U.S. Patent No. 5,715,403 **Extrinsic:** 1. Container: "contains protected *content*, which is divided into one or more *atomic elements*, and, optionally, *PERCs* governing the *content* and may be manipulated only as specified by a *PERC.*" (IT Glossary, 4/6/95, IT00028206) 2. Container: "A packaging mechanism, consisting of: *One or more Element-derived components. *An organization mechanism which provides a unique name within a flat namespace for each of the components in a Container." (IT Glossary, 5/12/95, IT00028293) 3. Container: "A protected digital information storage and transport mechanism for packaging content and control information." (IT Glossary, 8/21/95, TD00068B, IT00032372) 4. Container: "A collection of content and control-related information." (IT VDE Container Overview, 2/10/95, ETM-9999 Version 0.21, IT00051228) 5. Container: "A dynamic data structure, the elements of which are arbitrary data items whose type is not known when the program is written." (Que) 6. Container: "Abstract data type storing a collection of objects (elements)." (Laplante) 7. See also IT00037-44, IT002734-39, IT004188-96, IT0031572-85, IN00075960, IT00703055-71, IT0052146-64, IN00441189-224, IN0075983-87 8. Contain: "In data security, a multilevel information structure. A container has a classification and may contain objects and/or other containers." (Longley) 9. U.S. Patent No. 5,369,702 10. See also Microsoft PLR 4-2 Exhs. E & F as revised, and InterTrust's Rule 30(b)(6) testimony.

	Claim Term/Phrase	Evidence Supporting MS Construction
8.	control (n.) 193.1, 193.11, 193.15, 193.19 683.2 891.1	**Intrinsic:** 1. "Claims ... are allowable over the prior art of record. The instant claims provide for first and second entity or control or procedure or executable code that are separately, remotely and different from each to combine or process or execute an operation or procedure based on at least first and second control or procedure or executable code in an electronic appliance or secure operating environment or third party different and remote from the first and second entity or control or procedure or executable code." (08/964,333 Patent Application Prosecution History, Office Action, 9/22/98, p. 3 (MSI028945)) 2. "The virtual distribution environment 100 prevents use of protected information except as permitted by the 'rules and controls' (control information)." ('193 56:26-28) 3. "As mentioned above, virtual distribution environment 100 'associates' content with corresponding 'rules and controls,' and prevents the content from being used or accessed unless a set of corresponding 'rules and controls' is available." ('193 57:18-22) 4. "...at least one rule and/or control associated with the software agent that governs the agent's operation." ('193 241:2-3) 5. "In this example control information may include one or more component assemblies that describe the articles within such a container (e.g. one or more event methods referencing map tables and/or algorithms that describe the extent of each article)." ('193 309:5-9) 6. "Even if a consumer has a copy of a video program, she cannot watch or copy the program unless she has 'rules and controls' that authorize use of the program. She can use the program only as permitted by the 'rules and controls.'" ('193 53:60-63) 7. "A control set 914 contains a list of required methods that must be used to exercise a specific right (i.e., process events associated with a right)." ('193 151:14-16) 8. "If necessary, trusted go-between 4700 may obtain and register any methods, rules and/or controls it needs to use or manipulate the object 300 and/or its contents (FIG. 122 block 4778)." ('683, 47:42-45) 9. "These rights govern use of the VDE object 300 by that user or user group. For instance, the user may have an 'access' right, and an 'extraction' right, but not a 'copy' right." ('193 159:23-26) 10. "To provide for this, ROS 602 may include a 'redirector' 684 that allows such 'non-VDE aware' applications 608(b) to access VDE objects 300 and functions 604. Redirector 684, in the preferred embodiment, translates OS calls directed to the 'other OS functions' 606 into calls to the 'VDE functions' 604. As one simple example, redirector 684 may intercept a 'file open' call from application 608(b), determine whether the file to be opened is contained within a VDE container 300, and if it is, generate appropriate VDE function call(s) to file system 687 to open the VDE container (and potentially generate events to HPE 655 and/or SPE 503 to determine the name(s) of file(s) that may be stored in a VDE object 300, establish a control structure associated with a VDE object 300, perform a registration for a VDE object 300, etc.). Without redirector 684 in this example, a

	Claim Term/Phrase	Evidence Supporting MS Construction
		non-VDE aware application such as 608b could access only the part of API 682 that provides an interface to other OS functions 606, and therefore could not access any VDE functions." ('193 82:27-45) 11. "An executing process cannot access memory outside its domain and can only communicate with other processes through services provided by and mediated by privileged kernel/dispatcher software 552 within the SPU 500." ('193 109:53-57) 12. "An electronic appliance 600 may not access an object unless a corresponding PERC 808 is present, and may only use the object and related information as permitted by the control structures contained within the PERC." ('193 118:17-31) 13. "Load modules are not necessarily directly governed by PERCs 808 that control them, nor must they contain any time/date information or expiration dates. The only control consideration in the preferred embodiment is that one or more methods 1000 reference them using a correlation tag (the value of a protected object created by the load module's owner, distributed to authorized parties for inclusion in their methods, and to which access and use is controlled by one or more PERCs 808). If a method core 1000' references a load module 1100 and asserts the proper correlation tag (and the load module satisfies the internal tamper checks for the SPE 503), then that load module can be loaded and executed, or it can be acquired from, shipped to, updated, or deleted by, other systems." (193 139:60 - 140:6) 14. "In the preferred embodiment, SPE RPC manager 550 first references a service request against the RPC service table to determine the location of the service manager that may service the request. The RPC manager 550 then routes the service request to the appropriate service manager for action. Service requests are handled by the service manager within the SPE 503 using the RPC dispatch table to dispatch the request. Once the RPC manager 550 locates the service reference in the RPC dispatch table, the load module that services the request is called and loaded using the load module execution manager 568. The load module execution manager 568 passes control to the requested load module after performing all required context configuration, or if necessary may first issue a request to load it from the external management files 610." ('193 148:55-58) 15. "Although methods 1000 can have virtually unlimited variety and some may even be user-defined, certain basic 'use' type methods are preferably used in the preferred embodiment to control most of the more fundamental object manipulation and other functions provided by VDE 100. For example, the following high level methods would typically be provided for object manipulation: OPEN method, READ method, WRITE method, CLOSE method. An OPEN method is used to control opening a container so its contents may be accessed. A READ method is used to control the access to contents in a container. A WRITE method is used to control the insertion of contents into a container. A CLOSE method is used to close a container that has been opened." ('193 183:12-29) 16. "FIG. 54 is a flowchart of an example of program control steps performed by an ACCESS method 2000. As described above, an ACCESS method may be used to access content embedded in an object 300 so it can be written to, read from, or

	Claim Term/Phrase	Evidence Supporting MS Construction
		otherwise manipulated or processed. In many cases, the ACCESS method may be relatively trivial since the object may, for example, be stored in a local storage that is easily accessible. However, in the general case, an ACCESS method 2000 must go through a more complicated procedure in order to obtain the object. For example, some objects (or parts of objects) may only be available at remote sites or may be provided in the form of a real-time download or feed (e.g., in the case of broadcast transmissions). Even if the object is stored locally to the VDE node, it may be stored as a secure or protected object so that it is not directly accessible to a calling process. ACCESS method 2000 establishes the connections, routings, and security requisites needed to access the object. These steps may be performed transparently to the calling process so that the calling process only needs to issue an access request and the particular ACCESS method corresponding to the object or class of objects handles all of the details and logistics involved in actually accessing the object." ('193 188:59-67) 17. "The READ control method 1652 must determine which key to use to decrypt content if it is going to release decrypted content to the user (block 1758). READ control method 1652 may make this key determination based, in part, upon the PERC 808 for the object (block 1760). READ control method 1652 may then call an ACCESS method to actually obtain the encrypted content to be decrypted (block 1762). The content is then decrypted using the key determined by block 1758 (block 1764)." ('193 192:2-24) 18. See also prior art referred to in the relevant InterTrust patent file histories, e.g., references made at the following bates ranges: MSI026598-602, MSI26626-7, MSI26630-42; MSI028808-11, MSI28846-52, MSI28728-62, MSI28857-58, MSI28944-97, MSI28953-56 19. "C_C may further include, for example: (a) a requirement that distributors ensure that creator C receive $1 per article accessed by users and/or user/distributors, which payment allows a user to access such an article for a period of no more than six months (e.g. using a map-type meter method that is aged once per month, time aged decryption keys, expiration dates associated with relevant permissions records, etc." ('193 309:10-16) 20. "It also employs a software object architecture for VDE content containers that carries protected content and may also carry both freely available information (e.g, summary, table of contents) and secured content control information which ensures the performance of control information." ('193 15:41-46) 21. "Because of the breadth of issues resolved by the present invention, it can provide the emerging 'electronic highway' with a single transaction/distribution control system that can, for a very broad range of commercial and data security models, ensure against unauthorized use of confidential and/or proprietary information and commercial electronic transactions." ('193 17:22-28) 22. "... (as allowed, or not prevented, by senior control information)." ('193 303:67 - 304:1) 23. "For purposes of expedition, applicants are rewriting these dependent claims into independent form, ... In addition, applicants have ... replaced 'necessary in order to gain' with 'allowing' in now-cancelled claim 204 incorporated into formerly dependent claims 209 & 21 1 [issued claim 35] ...." (Prosecution

	Claim Term/Phrase	Evidence Supporting MS Construction
		History for the 08/780,545 Patent Application (issued as the '912), Amendment, 10/29/98) 24. "VDE can: (a) audit and analyze the use of content, (b) ensure that content is used only in authorized ways, and (c) allow information regarding content usage to be used only in ways approved by content users." ('193 4:51-56) 25. "VDE is a secure system for regulating electronic conduct and commerce. Regulation is ensured by control information put in place by one or more parties." ('193 6:33-34) 26. "VDE ensures that certain prerequisites necessary for a given transaction to occur are met." ('193 20:27-28) **Extrinsic:** 1. Control: "The determination of the time and order in which the parts of a data processing system and the devices that contain those parts perform the input, processing, storage, and output functions." (IBM) 2. "5. Control Notes ... A Control must execute as a transaction ... A Control may require pre-conditions – that is that one or more other Controls have been executed before the Control is executed. ... 7. Control Execution Flow The following pseudocode describes the approximate execution sequence for a View Control ... 8. Operation of a Control (Execution of 'Rules and Consequences') ..." (VDE Controls Notes, IT00051953-55) 3. Control: "A business rule that governs the use of content." (IT Glossary, 1997-1998, ML00012B) 4. Control: "A set of rules and consequences that apply to a governed element. The term control can apply to either a control program or a control set." (IT Glossary, 1997-2000, ML00012D) 5. Control: "**Control Element*: A data structure that giverns [sic] the operation of a control mechanism (e.g., meter element, budget element, report element, trail element). **Control mechanism*: One of the mechanisms that controls and performs operations on a VDE object (e.g. meter, bill, budget). A control mechanism is distinct from a control element in that it specifies the execution of some process. * *Control object:* A data structure that is used to implement some VDE control: a PERC, a control element, a control parameter, or the data representing a control mechanism. **Control Parameter:* A data structure that is input to a control mechanism and that serves as part of the mechanism's specifications. For example, a billing mechanism might have a pricing parameter; a creator using that mechanism could alter the parameter but not change the mechanism itself." (IT Glossary, 3/7/95, IT00709618) 6. Control: "Defines rules and consequences for operations on a Property Chunk. A Control may be implemented by a process of arbitrary complexity (within the limits posed by the capability of the Node." (IT Glossary, 5/12/95, IT00028293) 7. Control: "A set of rules and consequences for operations on content, such as pricing, payment models, usage reporting etc." (IT Glossary, 8/21/95, TD00068B, IT00032373) 8. Control: "An object of the InterTrust Commerce Architecture that specifies business rules. Controls are applied at any time and at any point in the Chain of

	Claim Term/Phrase	Evidence Supporting MS Construction
		Handling and Control. InterTrust controls are dynamic, independent, and persistent." (IT Glossary, 11/17/96, TD00189J, IT00035865) 9. ""'Rules and Controls' means any electronic information that directs, enables, specifies, describes, and/or provides contributing means for performing or not-performing, permitted and/or required operations related to Content, including, for example, restricting or otherwise governing the performance of operations, such as, for example, Management of such Content." (License Agreement, InterTrust/Universal Music Group, 4/13/99, Exhibit 11 to InterTrust 30(b)(6)) 10. "A set of control elements corresponding to all of the property elements of a property. There may be zero or more controls for a given property." (IT 0028204) 11. "CONTROL(S): Controls refer to the rules and consequences associated with DigiBox containers. Controls may be applied dynamically..." (IT00035961) 12. "CONTROL: The rules associated with a governed entity such as a DigiBox container, property, or another control ... applied dynamically. InterTrust controls are dynamic, independent, and persistent." (IT00035920) 13. "... controls implement business rules..." (IT00035892) 14. "The function of performing required operations when certain specific conditions occur or when interpreting and acting upon instructions." (Webster's) 15. Access (n.): "2. The use of an access method. 3. The manner in which files or data sets are referred to by the computer. ... 5. In computer security, a specific type of interaction between a subject and an object that results in the flow of information from one to the other." (IBM) 16. Access (n.): "1. In access control, a specific type of interaction between a subject and an object that results in the flow of information from one to the other ... 3. In computing, the manner in which files or data sets are referred to by a computer." (Longley) 17. Access(ing) (v.): "1. To obtain the use of a computer resource. ... 4. To obtain data from or to put data in storage." (IBM) 18. Least privilege: "Each user and each program should operate using the fewest privileges possible. In this way, the damage from an inadvertent or malicious attack is minimized." (Pfleeger) 19. See also IT00125, IT31410-14, IT703083-89, IT51721-26, IT00735936 (key), IT51956 et seq., IN0075983-87, IN0075989-93 20. See also Microsoft PLR 4-2 Exhs. E & F as revised, and InterTrust's Rule 30(b)(6) testimony.
9.	controlling, control (v.) 193.1 861.58	**Intrinsic:** 1. "ROS 602 includes software intended for execution by SPU microprocessor 520 for, in part, controlling usage of VDE related objects 300 by electronic appliance 600. As will be explained, these SPU programs include 'load modules' for performing basic control functions." ('193 66:5-8) 2. "VDE prevents many forms of unauthorized use of electronic information, by controlling and auditing (and other administration of use) electronically stored and/or disseminated information." ('193 11:60-63) 3. "It also employs a software object architecture for VDE content containers that carries protected content and may also carry both freely available information

	Claim Term/Phrase	Evidence Supporting MS Construction
		(e.g., summary, table of contents) and secured content control information which ensures the performance of control information." ('193 15:41-46) 4. "VDE ensures that certain prerequisites necessary for a given transaction to occur are met." ('193 20:27-28) 5. "The virtual distribution environment 100 prevents use of protected information except as permitted by the 'rules and controls' (control information)." ('193 56:26-28) 6. "As mentioned above, virtual distribution environment 100 'associates' content with corresponding 'rules and controls,' and prevents the content from being used or accessed unless a set of corresponding 'rules and controls' is available." ('193 57:18-22) 7. "VDE can: (a) audit and analyze the use of content, (b) ensure that content is used only in authorized ways, and (c) allow information regarding content usage to be used only in ways approved by content users." ('193 4:51-56) 8. "VDE is a secure system for regulating electronic conduct and commerce. Regulation is ensured by control information put in place by one or more parties." ('193 6:33-35) 9. "It also employs a software object architecture for VDE content containers that carries protected content and may also carry both freely available information (e.g., summary, table of contents) and secured content control information which ensures the performance of control information." ('193 15:41-46) 10. "Because of the breadth of issues resolved by the present invention, it can provide the emerging 'electronic highway' with a single transaction/distribution control system that can, for a very broad range of commercial and data security models, ensure against unauthorized use of confidential and/or proprietary information and commercial electronic transactions." ('193 17:22-28) 11. "VDE ensures that certain prerequisites necessary for a given transaction to occur are met." ('193 20:27-28) **Extrinsic:** 1. Control: "The determination of the time and order in which the parts of a data processing system and the devices that contain those parts perform the input, processing, storage, and output functions." (IBM) 2. Control: "A business rule that governs the use of content." (IT Glossary, 1997-1998, ML00012B) 3. Control: "A set of rules and consequences that apply to a governed element. The term control can apply to either a control program or a control set." (IT Glossary, 1997-2000, ML00012D) 4. Control: "**Control Element*: A data structure that giverns (*sic*) the operation of a control mechanism (e.g., meter element, budget element, report element, trail element). **Control mechanism*: One of the mechanisms that controls and performs operations on a VDE object (e.g. meter, bill, budget). A control mechanism is distinct from a control element in that it specifies the execution of some process. * *Control object:* A data structure that is used to implement some VDE control: a PERC, a control element, a control parameter, or the data representing a control mechanism. **Control Parameter:* A data structure that is

	Claim Term/Phrase	Evidence Supporting MS Construction
		input to a control mechanism and that serves as part of the mechanism's specifications. For example, a billing mechanism might have a pricing parameter; a creator using that mechanism could alter the parameter but not change the mechanism itself." (IT Glossary, 3/7/95, IT00709618) 5. Control: "Defines rules and consequences for operations on a Property Chunk. A Control may be implemented by a process of arbitrary complexity (within the limits posed by the capability of the Node." (IT Glossary, 5/12/95, IT00028293) 6. Control: "A set of rules and consequences for operations on content, such as pricing, payment models, usage reporting etc." (IT Glossary, 8/21/95, TD00068B, IT00032373)
10.	copy, copied, copying 193.1, 193.11, 193.15, 193.19	**Intrinsic**: 1. "These rights govern use of the VDE object 300 by that user or user group. For instance, the user may have an 'access' right, and an 'extraction' right, but not a 'copy' right." ('193 159:23-26) 2. "At the same time, electronic testing will allow users to receive a copy (encrypted or unencrypted) of their test results when they leave the test sessions." ('193 319:12-15) 3. "This is because VDE objects may contain data that can be electronically copied outside the confines of a VDE node. If the content is encrypted, the copies will also be encrypted and the copier cannot gain access to the content unless she has the appropriate decryption key(s)." ('193 129:3-8) 27. "Even if a consumer has a copy of a video program, she cannot watch or copy the program unless she has 'rules and controls' that authorize use of the program. She can use the program only as permitted by the 'rules and controls.'" ('193 53:60-63) 4. "For example, if a software program was distributed as a traveling object, a user of the program who wished to supply it or a usable copy of it to a friend would normally be free to do so." ('193 131:65 - 132:1) 5. "Storing a first digital file and a first control in a first secure container, said first control constituting a first budget which governs the number of copies which may be made of said first digital file or a portion of said first digital file while said first digital file is contained in said first secure container." ('193 330:1 -331:25 (claim 60)) **Extrinsic:** 1. Copy: "A product of a document copying process." (IBM)
11.	derive 900.155	**Intrinsic**: 1. "Such control information can continue to manage usage of container content if the container is 'embedded' into another VDE managed object, such as an object which contains plural embedded VDE containers, each of which contains content derived (extracted) from a different source." ('193 28:60-65)
12.	designating 721.1	

	Claim Term/Phrase	Evidence Supporting MS Construction
13.	device class 721.1	**Intrinsic:** 1. "Furthermore, Applicants respectfully submit that some of the terms cited by the Examiner as 'indefinite' are either well-known by persons skilled in the art or inherently clear. For example, in Claims 1-4, 22-25, the term 'class' is used as part of the phrase 'device class.' Applicants respectfully submit that 'device class' is inherently clear, meaning a group of devices which share at least one attribute." (Prosecution History for the 08/689,754 Patent Application (issued as the '721), Amendment, 4/14/99, p. 14) **Extrinsic:** 1. Device: "1. A mechanical, electrical, or electronic contrivance with a specific purpose." (IBM) 2. Device class: "The generic name for a group of device types." (IBM) 3. Device type: "1. The name for a kind of device sharing the same model number; for example, 2311, 2400, 2400-1. Contrast with device class. 2. The generic name for a group of devices; for example, 5219 for IBM 5219 Printers. Contrast with device class." (IBM)
14.	digital signature, digitally signing 721.1	**Intrinsic:** 1. "There exist many well known processes for creating digital signatures. One example is the Digital Signature Algorithm (DSA). DSA uses a public-key signature scheme that performs a pair of transformations to generate and verify a digital value called a 'signature.'" ('721 10:60-64) 2. "A verifying authority digitally 'signs' and 'certifies' those load modules or other executables it has verified (using a public key based digital signature and/or certificate based thereon, for example)." ('721 4:64-67) 3. "The algorithm also makes use of a one-way hash function, H(m), such as, for example, the Secure Hash Algorithm. The first three parameters, p, q, and g, are public and may be shared across a network of users. The private key is x; the public key is y. To sign a message, m, using DSA, a signer generates a random number, k, less than q. The signer also generates: $r=(g^k \bmod p) \bmod q$; and $s=(k^{-1}(H(m)+xr)) \bmod q$. The parameters r and s comprise the signer's signature, which may be sent to a recipient or distributed across a network." ('721 11:7-22) 4. "Protected processing environment 108 then decrypts digital signature 106 using the second key 124--i.e., it opens strongbox 118 to retrieve the message digest 116 a verifying authority 100 placed in there. Protected processing environment 108 compares the version of message digest 116 it obtains from the digital signature 106 with the version of message digest 116' it calculates itself from load module 54 using the one way hash transformation 115. The message digests 116, 116' should be identical. If they do not match, digital signature 106 is not authentic or load module 54 has been changed--and protected processing environment 108 rejects load module 54." ('721 14:49-60) 5. "One digital signature 106(1) can be created by encrypting message digest 116 with a 'private' key 122(1), another (different) digital signature 106(2) can be created by encrypting the message digest 116 with a different 'private' key 122(2), possibly employing a different signature algorithm." ('721 14:64 - 15:2) 6. "Certificates play an important role in the trustedness of digital signatures, and

	Claim Term/Phrase	Evidence Supporting MS Construction

also are important in the public-key authentication communications protocol (to be discussed below). In the preferred embodiment, these certificates may include information about the trustedness/level of security of a particular VDE electronic appliance 600 (e.g., whether or not it has a hardware-based SPE 503 or is instead a less trusted software emulation type HPE 655) that can be used to avoid transmitting certain highly secure information to less trusted/secure VDE installations." ('193 203:58-67)

7. "Master Keys: A 'master' key is a key used to encrypt other keys. An initial or 'master' key may be provided within PPE 650 for communicating other keys in a secure way. During initialization of PPE 650, code and shared keys are downloaded to the PPE. Since the code contains secure convolution algorithms and/or coefficients, it is comparable to a 'master key.' The shared keys may also be considered 'master keys.'" ('193 212:12-18)
8. "FIGS. 64 through 67 illustrate the preferred public-key embodiment, but may also be used to help understand the secret-key versions. In secret-key embodiments, the certification process and the public key encryptions/decryptions are replaced with private-key encryptions, and the public key/private-key pairs are replaced with individual secret keys that are shared between the PPE 650 instance and the other parties (e.g., the load module supplier(s), the PPE manufacturer). In addition, the certificate generation process 2804 is not performed in secret-key embodiments, and no site identity certificates 2823 or VDE certificate database 2830 exist." ('193 211:18-30)
9. "Key Types
   The detailed descriptions of key types below further explain secret-key embodiments; this summary is not intended as a complete description. The preferred embodiment PPE 650 can use different types of keys and/or different 'shared secrets' for different purposes. Some key types apply to a Public-Key/Secret Key implementation, other keys apply to a Secret Key only implementation, and still other key types apply to both. The following table lists examples of various key and 'shared secret' information used in the preferred embodiment, and where this information is used and stored:

Key/Secret Information Type	Used in PK or Non-PK	Example Storage Location(s)
Master Key(s) (may include some of the specific keys mentioned below)	Both	PPE Manufacturing facility VDE administrator
Manufacturing Key	Both (PK optional)	PPE (PK case) Manufacturing facility
Certification key pair	PK	PPE Certification repository
Public/private key pair	PK	PPE Certification repository (Public Key only)
Initial secret key	Non-PK	PPE
PPE manufacturing ID	Non-PK	PPE
Site ID, shared code, shared keys and shared secrets	Both	PPE
Download authorization key	Both	PPE

	Claim Term/Phrase	Evidence Supporting MS Construction

			VDE administrator
External communication keys and other info		Both	PPE Secure Database
Administrative object keys		Both	Permission record
Stationary object keys		Both	Permission record
Traveling object shared keys		Both	Permission record
Secure database keys		Both	PPE
Private body keys		Both	Secure database Some objects
Content keys		Both	Secure database Some objects
Authorization shared secrets		Both	Permission record
Secure Database Back up keys		Both	PPE Secure database"

('193 211:31 – 212:11)

10. "The process for this selection is similar to the process used by EVENT methods to map events into atomic element numbers. DECRYPT method 2030 may then access an appropriate PERC 808 from the secure database 610 and loads a key (or 'seed') from a PERC (blocks 2034, 2036). This key information may be the actual decryption key to be used to decrypt the content, or it may be information from which the decryption key may be at least in part derived or calculated. If necessary, DECRYPT method 2030 computes the decryption key based on the information read from PERC 808 at block 2034 (block 2038). DECRYPT method 2030 then uses the obtained and/or calculated decryption key to actually decrypt the block of encrypted information (block 2040). DECRYPT method 2030 outputs the decrypted block (or the pointer indicating where it may be found), and terminates (termination point 2042)." ('193 193:8-23)
11. "A 'time aged key' in the preferred embodiment is not a 'true key' that can be used for encryption/decryption, but rather is a piece of information that a PPE 650, in conjunction with other information, can use to generate a 'true key.' This other information can be time-based, based on the particular 'ID' of the PPE 650, or both. Because the 'true key' is never exposed but is always generated within a secure PPE 650 environment, and because secure PPEs are required to generate the 'true key,' VDE 100 can use 'time aged' keys to significantly enhance security and flexibility of the system." ('193 207:50-60)
12. "Running the function with a time-aged key and inappropriate time values typically yields a useless key that will not decrypt." ('193 208:38-40)

Extrinsic:

1. Digital Signature: "In computer security, encrypted data, appended to or part of a message, that enables a recipient to prove the identity of the sender." (IBM)
2. Digital Signature: "1. In authentication, data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery. 2. In authentication, a data block appended to a message, or a complete encrypted message, such that the recipient can authenticate the message contents and/or prove that it could only have originated with the purported sender." (Longley)

	Claim Term/Phrase	Evidence Supporting MS Construction
		3. "Let B be the recipient of a message M signed by A, then A' s [digital] signature must satisfy three requirements: B must be able to validate A's signature on M. It must be impossible for anyone, including B, to forge A's signature. In case A should disavow signing a message M, it must be possible for a judge or third party to resolve a dispute arising between A and B. A digital signature therefore establishes sender authenticity ..., it also establishes data authenticity." (Denning, p. 14) 4. "A cipher is unconditionally secure if, no matter how much ciphertext is intercepted, there is not enough information in the ciphertext to determine the plaintext uniquely." (Denning, p. 5) (Davies, pp. 41, 380) 5. "A cipher is computationally secure, or strong, if it cannot be broken by systematic analysis with available resources." (Denning, p. 5) (Davies, pp. 41, 370) 6. Key: "7. In computer security, a sequence of symbols used with a cryptographic algorithm for encrypting or decrypting data." (IBM) 7. Key: "1. In cryptography, a sequence of symbols that controls the operations of encipherment and decipherment. 2. In cryptography, a symbol or sequence of symbols (or electrical or mechanical correlates of symbols) that control the operations of encryption and decryption)." (Longley)
15.	executable programming, executable 721.34 912.8, 912.35	**Intrinsic**: 1. "Furthermore, applicants' independent claims 16, 36, 37 and 64 require secure delivery and use of plural executable items. See claim 16 ('securely delivering a first procedure ... securely delivering ... a second procedure separable or separate from said first procedure...'); claim 36 ('securely delivering plural executable procedures ...'), claim 37 ('securely delivering a first piece of executable code ... securely delivering a second piece of executable code ...') and claim 64 ('securely receiving a first load module ... securely receiving a second load module ...'). These features are not taught or suggested by either Rosen or Johnson. Johnson's databases comprise data, not executable code." (Prosecution for the 08/388,107, Patent Application, Amendment, 6/20/97, pp. 24-25) (MSI028848-49) 2. "In addition, Applicants would like to draw the Examiner's attention to other sections of the specification in support of words or phrases cited by the Examiner as 'indefinite.' ... The noun 'executable,' as used in Claims ... 34-36 ..., is defined in the specification on page 7." (Prosecution History for the 08/689,754 Patent Application (issued as the '721 patent), Amendment, 4/14/99, pp. 13-14) (p. 7 of the original specification is '721 2:62 - 3:13 of the issued patent) **Extrinsic:** 1. Execute: "1. To perform the actions specified by a program or a portion of a program." (IBM) 2. Executable Program: "1. A program that has been link-edited and therefore can be run in a processor. 2. The set of machine language instructions that constitute the output from the compilation of a source program." (IBM)

	Claim Term/Phrase	Evidence Supporting MS Construction
16.	host processing environment 900.155	**Intrinsic:** 1. "Portions of ROS 602 in particular may desirably be included in ROM 658 (e.g., 'bootstrap' routines, POST routines, etc. for use in establishing an operating environment for electronic appliance 600 when power is applied)." ('193 63:13-17) 2. "In the preferred embodiment, HPE 655 is a secure processing environment supported by a processor other than an SPU, such as for example an electronic appliance CPU 654 general-purpose microprocessor or other processing system or device. In the preferred embodiment, HPE 655 may be considered to 'emulate' an SPU 500 in the sense that it may use software to provide some or all of the processing resources provided in hardware and/or firmware by an SPU." ('193 79:60-67) 3. "However, in applications where lesser security can be tolerated and/or the cost of an SPU 500 cannot be tolerated, the SPE 503 may be omitted and all secure processing may instead be performed by one or more secure HPEs 655 executing on general- purpose CPUs 654." ('193 81:4-8) 4. "Integrity of Software-Based PPE Security: As discussed above in connection with FIG. 10, some applications may use a software-based protected processing environment 650 (such as a 'host event processing environment' (HPE) 655) providing a software-based tamper resistant barrier 674." ('900 230:57-61) 5. "In one example, the software distribution medium 3370 might include installation materials 3470 and operational materials 3472. The installation materials 3470 may be executed by computer 3372 to install the operational materials 3472 onto the computer's hard disk 3376. The computer 3372 may then execute the operational materials 3472 from its hard disk 3376 to provide software-based protected processing environment 650 and associated software-based tamper resistant barrier 672." ('900 231:23-31) 6. "The operational materials 3472 may provide executable code and associated data structures for providing protected processing environment 650 and associated software-based tamper resistant barrier 674." ('900 236:50-53) 7. "HPE(s) 655 and SPE(s) 503 are self-contained computing and processing environments that may include their own operating system kernel 688 including code and data processing resources." ('193 79:36-39) 8. "HPEs 655 may be provided in two types: secure and not secure." ('193 80:8-9) 9. "[T]his example also includes one or more Host Event Processing Environments ('HPEs') 655 and/or one or more Secure Event Processing Environment ('SPEs') 503 (these environments may be generically referred to as 'Protected Processing Environments' 650)." ('193 79:31-35) 10. "HPEs 655 may (as shown in FIG. 10) be provided with a software- based tamper resistant barrier 674 that makes them more secure. Such a software-based tamper resistant barrier 674 may be created by software executing on general-purpose CPU 654. Such a 'secure' HPE 655 can be used by ROS 602 to execute processes that, while still needing security, may not require the degree of security provided by SPU 500. This can be especially beneficial in architectures providing both an SPE 503 and an HPE 655. The SPU 502 may be used to perform all truly secure processing, whereas one or more HPEs 655 may be used to provide additional

	Claim Term/Phrase	Evidence Supporting MS Construction
		secure (albeit possibly less secure than the SPE) processing using host processor or other general purpose resources that may be available within an electronic appliance 600. Any service may be provided by such a secure HPE 655" ('193 80:22-36) 11. "The software-based tamper resistant barrier 674 provided by HPE 655 may be provided, for example, by: introducing time checks and/or code modifications to complicate the process of stepping through code comprising a portion of kernel 688a and/or a portion of component assemblies 690 using a debugger; using a map of defects on a storage device (e.g., a hard disk, memory card, etc.) to form internal test values to impede moving and/or copying HPE 655 to other electronic appliances 600; using kernel code that contains false branches and other complications in flow of control to disguise internal processes to some degree from disassembly or other efforts to discover details of processes; using 'self-generating' code (based on the output of a co-sine transform, for example) such that detailed and/or complete instruction sequences are not stored explicitly on storage devices and/or in active memory but rather are generated as needed; using code that 'shuffles' memory locations used for data values based on operational parameters to complicate efforts to manipulate such values; using any software and/or hardware memory management resources of electronic appliance 600 to 'protect' the operation of HPE 655 from other processes, functions, etc. Although such a software-based tamper resistant barrier 674 may provide a fair degree of security, it typically will not be as secure as the hardware-based tamper resistant barrier 502 provided (at least in part) by SPU 500." ('193 80:40-65; Fig. 10) 12. "FIG. 12 also shows that ROS 602 may provide one or more SPEs 503 and/or one or more HPEs 655. As discussed above, HPE 655 may 'emulate' an SPU 500 device, and such HPEs 655 may be integrated in lieu of (or in addition to) physical SPUs 500 for systems that need higher throughput. Some security may be lost since HPEs 655 are typically protected by operating system security and may not provide truly secure processing Thus, in the preferred embodiment, for high security applications at least, all secure processing should take place within SPE 503 having an execution space within a physical SPU 500 rather than a HPE 655 using software operating elsewhere in electronic appliance 600." ('193 88:31-43) 13. "As discussed above in connection with FIG. 12, each electronic appliance 600 in the preferred embodiment includes one or more SPEs 503 and/or one or more HPEs 655. These secure processing environments each provide a protected execution space for performing tasks in a secure manner." ('193 104:39-44) **Extrinsic:** 1. Host processor: "1. A processor that controls all or part of a user application network. 2. In a network, the processing unit in which resides the access method for the network. ... 4. A processing unit that executes the access method for attached communication controllers." (IBM) 2. "Host Processing Environment (HPE): A software-only realization of the PPE, protected from tampering by appropriate software techniques. No longer preferred because of the potential confusion between the 'H' in the acronym and

	Claim Term/Phrase	Evidence Supporting MS Construction
		'H' as in 'Hardware' (which this isn't). [REPLACEMENT UNCERTAIN]" (IT Glossary, "Obsolete Terminology Section,"[2] 3/7/95, IT00709621) 3. "Secure Processing Environment (SPE): A hardware-supported realization of the PPE, protected from tampering by physical security techniques. No longer preferred because of the potential confusion between the 'S' in the acronym and 'S' as in 'Software' (which this isn't). [REPLACEMENT UNCERTAIN]" (IT Glossary, "Obsolete Terminology Section" 5/12/95, IT00028302) 4. Environment: See InterTrust node: "A computer that is enabled for processing of DigiBox containers by installation of a PPE, which may be either hardware or software based. A node may include application software and/or operating system integration. The node is also termed the *environment*." (IT Glossary, 8/21/95, TD00068B, IT00032375)
17.	identifier 193.15 912.8	**Intrinsic:** 1. "Portable appliance 2600 RAM 534 may contain, for example, information which can be used to uniquely identify each instance of the portable appliance. This information may be employed (e.g. as at least a portion of key or password information) in authentication, verification, decryption, and/or encryption processes." ('193 230:22-27) 2. "Provide very flexible and extensible user identification according to individuals, installations, by groups such as classes, and by function and hierarchical identification employing a hierarchy of levels of client identification (for example, client organization ID, client department ID, client network ID, client project ID, and client employee ID, or any appropriate subset of the above)." ('193 25:31-38) 3. "Fingerprinting is useful in providing an ability to identify who extracted information in clear form *[sic]* a VDE container, or who made a copy of a VDE object or a portion of its contents." ('193 37:27-31) 4. "All load modules 1100 for use by SPE 503 are preferably referenced by a load module execution manager 568 that maintains and scans a list of available load modules and selects the appropriate load module for execution. If the load module is not present within SPE 503, the task is 'slept' and LMEM 568 may request that the load module 1100 be loaded from secondary storage 562. This request may be in the form of an RPC call to secure database manager 566 to retrieve the load module and associated data structures, and a call to encrypt/decrypt manager 556 to decrypt the load module before storing it in memory allocated by memory manager 578." ('193 111:47-58) 5. "In somewhat more detail, the preferred embodiment executes a load module 1100 by passing the load module execution manager 568 the name (e.g., VDE ID) of the desired load module 1100. LMEM 568 first searches the list of 'in memory' and 'built-in' load modules 572. If it cannot find the desired load

[2] Some terms were "defined" in an "Obsolete Terminology Section" of certain IT Glossaries. This section was described in such documents as: "This section identifies terms that have been used in earlier documents to describe various VDE concepts, but that are, for various reasons, no longer preferred." (See, e.g., IT00028302)

	Claim Term/Phrase	Evidence Supporting MS Construction
		module 1100 in the list, it requests a copy from the secure database 610 by issuing an RPC request that may be handled by ROS secure database manager 744 shown in FIG. 12." ('193 111:59-67) 6. "For each VDE item loaded into SPE 503, Secure Database manager 566 in the preferred embodiment may search a master list for the VDE item ID, and then check the corresponding transaction tag against the one in the item to ensure that the item provided is the current item. Secure Database Manager 566 may maintain list of VDE item ID and transaction tags in a 'hash structure' that can be paged into SPE 503 to quickly locate the appropriate VDE item ID. In smaller systems, a look up table approach may be used. In either case, the list should be structured as a pagable *[sic]* structure that allows VDE item ID to be located quickly." ('193 124:8-18) 7. "A stipulation that the traveling object may be used on certain one or more installations or installation classes or users or user classes where classes correspond to a specific subset of installations or users who are represented by a predefined class identifiers stored in a secure database 610." ('193 131:40-45) 8. "A load module 1100 is able to perform its function only when executed in the protected environment of an SPE 503 or an HPE 655 because only then can it gain access to the protected elements (e.g., UDEs 1200, other load modules 1100) on which it operates. Initiation of load module execution in this environment is strictly controlled by a combination of access tags, validation tags, encryption keys, digital signatures and/or correlation tags. Thus, a load module 1100 may only be referenced if the caller knows its ID and asserts the shared secret correlation tag specific to that load module. The decrypting SPU may match the identification token and local access tag of a load module after decryption. These techniques make the physical replacement of any load module 1100 detectable at the next physical access of the load module." ('193 139:41-55) 9. "These shared secrets may be used during communications processes to permit PPEs 650 to authenticate the identity of other PPEs and/or users." ('193 214:39-41) 10. "As another example, interpreter 508 may provide application 506 with an element identification (e.g., a hexadecimal value or other identifier) that corresponds to the headline information within the newspaper style content (block 558). Application 506 may then ask electronic appliance 500 to provide it with the Headline (or other) content information 102 within container 100 by providing appropriate content information to electronic appliance 500 via APL 504 (block 560)." ('861 12:63 - 13:4) 11. "It is preferable that an extremely secure encryption/decryption technique be used as an aspect of authenticating the identity of electronic appliances 600 that are establishing a communication channel and securing any transferred permission, method, and administrative information." ('193 67:21-26) 12. "As part of the initialization process, the PPE 650 may generate internally or the manufacturer may generate and supply, one or more pairs of site-specific public keys 2815 and private keys 2816. These are used by the PPE 650 to prove its identity." ('193 209:63-67)

	Claim Term/Phrase	Evidence Supporting MS Construction
		Extrinsic: 1. Identifier: "1. One or more characters used to identify or name a data element and possibly to indicate certain properties of that data element. 2. In programming languages, a token that names a data object such as a variable, an array, a record, a subprogram or a function." (IBM) 2. Identifier: "1. In computing, a character or group of characters used to identify, indicate or name a body of data. 2. In computing, a name or string of characters employed to identify a variable, procedure, data structure or some other element of a program." (Longley)
18.	protected processing environment 683.2 721.34	See also "secure" **Intrinsic:** 1. Prosecution History of Application 08/778,256 (continuation of '891 Patent, issued as U.S. Patent No. 5,949,876), Amendment, 1/20/98, pp. 58-60: a. "Independent claims 65 and 76 each recite a 'protected processing environment.' ... Griffeth et al. [U.S. Pat. No. 5,505,837], Yamamoto [U.S. Pat. No. 5,508,913] and Wyman [U.S. Pat. No. 5,260,999] do not disclose these aspects of these claims. b. The system disclosed in Griffeth et al is designed to allow negotiation to proceed in an environment in which a negotiating party does not disclose information about its negotiation goals to the other negotiating party. ... Griffeth et al. does not disclose any privacy protection mechanism and neither teaches nor suggests any secure processing environment or that any operations (e.g., integration or execution) occur securely. Indeed, Griffeth contains no suggestion that any protection mechanism is needed to maintain negotiation goals in privacy, since Griffeth does not suggest that the other party may try to improperly discover information which is intended to remain private. c. Yamamoto states the following: 'Here, the data is enciphered by the data encipher apparatuses 26 so as to maintain confidentiality.' Col. 3, lines 46-47. Since Yamamoto makes no other reference to the encipherment, or to the apparatuses 26, it is impossible to determine how the data encipherment is used, or the roles it plays in the disclosed apparatus. From an examination of Fig. 3, however, it appears that the data encipher apparatuses 26 are placed on connections between a particular site and other, physically separated sites. For example, customer office 23b is connected to sub-center 22 by a line, which apparently represents a communication path. That line connects directly to a data encipher apparatus 26 in customer office 23b, and to another data encipher apparatus 26 in sub-center 22. d. Thus, it appears that the data encipher apparatuses 26 are used, in some undisclosed manner, to encipher at least some data which travels among physically separated locations. It is possible to imagine, for example, that data is enciphered prior to being sent out on an insecure public transmission line, and is then deciphered once received in a new location. e. Yamamoto does not disclose, however, that the processing environments are themselves secure, or that either execution or integration occur in a secure manner or in a secure environment. Indeed, Yamamoto contains no suggestion that security within a processing environment would even be desirable. By

	Claim Term/Phrase	Evidence Supporting MS Construction
		suggesting that data is deciphered once it enters an office (e.g., office 23b), in fact, Yamamoto teaches away from a secure environment, since it would appear that the data is used 'in the clear' within the office, with no suggested protection beyond a simple password for the computer. f. Wyman is equally deficient regarding these elements. Although Wyman specifies that a license may contain a digital signature, therefore rendering the license unforgeable (Col. 14, lines 24-54), Wyman neither teaches nor suggests that the processing environment is itself secure or that any operations occur in a secure manner. The Wyman digital signatures no more suggest a secure processing environment than the requirement that paper contracts be signed in ink suggests that the contracts will be created, read or negotiated in a secure location." 2. "The role of go-between 4700 may, in some circumstances, be played by one of the participant's SPU's 500 (PPEs), since SPU (PPE) behavior is not under the user's control, but rather can be under the control of rules and controls provided by one or more other parties other than the user (although in many instances the user can contribute his or her own controls to operate in combination with controls contributed by other parties)." ('683 24:26-33) 3. "SPU 500 provides a tamper-resistant protected processing environment ("PPE") in which processes and transactions can take place securely and in a trusted fashion." ('683 16:60-62) 4. "The computer 3372 may then execute the operational materials 3472 from its hard disk 3376 to provide software-based protected processing environment 650 and associated software-based tamper resistant barrier 672)." ('900 231:27-31) 5. "The special purpose secure circuitry provided by the present invention includes at least one of: a dedicated semiconductor arrangement known as a Secure Processing Unit (SPU) and/or a standard microprocessor, microcontroller, and/or other processing logic that accommodates the requirements of the present invention and functions as an SPU." ('193 20:58-63) 6. "This means that a VDE SPU can employ (share) circuitry elements of a 'standard' CPU. For example, if a 'standard' processor can operate in protected mode and can execute VDE related instructions as a protected activity, then such an embodiment may provide sufficient hardware security for a variety of applications and the expense of a special purpose processor might be avoided." ('193 21:11-17) 7. "Different protected processing environments (secure execution spaces) might examine different subsets of the multiple digital signatures--so that compromising one protected processing environment (secure execution space) will not compromise all of them." ('721 7:19-23) 8. "The assurance level III appliance 61C shown is a general purpose personal computer equipped with a hardware-based secure processing unit 132 providing and completely containing protected processing environment 108 (see Ginter et al. FIGS. 6 and 9 for example). A silicon-based special purpose integrated circuit security chip is relatively more tamper-resistant than implementations relying on software techniques for some or all of their tamper-resistance." ('721 16:64 - 17:5)

	Claim Term/Phrase	Evidence Supporting MS Construction
		9. "FIG. 10 is a block diagram of one example of a software structure/architecture for Rights Operating System ('ROS') 602 provided by the preferred embodiment. In this example, ROS 602 includes an operating system ('OS') 'core' 679, a user Application Program Interface ('API') 682, a 'redirector' 684, an 'intercept' 692, a User Notification/Exception Interface 686, and a file system 687. ROS 602 in this example also includes one or more Host Event Processing Environments ('HPEs') 655 and/or one or more Secure Event Processing Environments ('SPEs') 503 (these environments may be generically referred to as 'Protected Processing Environments' 650). HPE(s) 655 and SPE(s) 503 are self-contained computing and processing environments that may include their own operating system kernel 688 including code and data processing resources." ('193 79:36-39) 10. "A given electronic appliance 600 may include any number of SPE(s) 503 and/or any number of HPE(s) 655. HPE(s) 655 and SPE(s) 503 may process information in a secure way, and provide secure processing support for ROS 602. For example, they may each perform secure processing based on one or more VDE component assemblies 690, and they may each offer secure processing services to OS kernel 680. In the preferred embodiment, SPE 503 is a secure processing environment provided at least in part by an SPU 500. Thus, SPU 500 provides the hardware tamper-resistant barrier 503 surrounding SPE 503. SPE 503 provided by the preferred embodiment is preferably: small and compact[,] loadable into resource constrained environments such as for example minimally configured SPUs 500[,] dynamically updatable[,] extensible by authorized users[,] integratable into object or procedural environments[, and] secure." ('193 79:39-59) 11. "As shown in FIG. 13, SPE 503 (PPE 650) includes the following service managers/major functional blocks in the preferred embodiment: Kernel/Dispatcher 552 Channel Services Manager 562 SPE RPC Manager 550 Time Base Manager 554 Encryption/Decryption Manager 556 Key and Tag Manager 558 Summary Services Manager 560 Authentication Manager/Service Communications Manager 564 Random Value Generator 565 Secure Database Manager 566 Other Services 592. Each of the major functional blocks of PPE 650 is discussed in detail below." ('193 105:23-41) 12. "I. SPE Kernel/Dispatcher: 552The Kernel/Dispatcher 552 provides an operating system 'kernel' that runs on and manages the hardware resources of SPU 500. This operating system 'kernel' 552 provides a self-contained operating system for SPU 500; it is also a part of overall ROS 602 (which may include multiple OS kernels, including one for each SPE and HPE ROS is controlling/managing). Kernel/dispatcher 552 provides SPU task and memory management, supports

	Claim Term/Phrase	Evidence Supporting MS Construction
		internal SPU hardware interrupts, provides certain 'low level services,' manages 'DTD' data structures, and manages the SPU bus interface unit 530. Kernel/dispatcher 552 also includes a load module execution manager 568 that can load programs into secure execution space for execution by SPU 500." ('193 105:43-57) (see also Fig. 13) 13. "In addition, memory management provided by memory manager 578 operating at least in part based on hardware-based MMU 540 may securely implement and enforce a memory architecture providing multiple protection domains. In such an architecture, memory is divided into a plurality of domains that are largely isolated from each other and share only specific memory areas under the control of the memory manager 578. An executing process cannot access memory outside its domain and can only communicate with other processes through services provided by and mediated by privileged kernel/dispatcher software 552 within the SPU 500. Such an architecture is more secure if it is enforced at least in part by hardware within MMU 540 that cannot be modified by any software-based process executing within SPU 500." ('193 109:46-60) 14. "Secure VDE hardware (also know as SPUs for Secure Processing Units), or VDE installations that use software to substitute for, or complement, said hardware (provided by Host Processing Environments (HPEs)), operate in conjunction with secure communications, system integration software, and distributed software control information and support structures, to achieve the electronic contract/rights protection environment of the present invention. Together, these VDE components comprise a secure, virtual, distributed content and/or appliance control, auditing (and other administration), reporting, and payment environment. In some embodiments and where commercially acceptable, certain VDE participants, such as clearinghouses that normally maintain sufficiently physically secure non-VDE processing environments, may be allowed to employ HPEs rather VDE hardware elements and interoperate, for example, with VDE end-users and content providers." ('193 13:7-23) 15. "Each PPE 650 needs to be initialized before it can be used. Initialization may occur at the manufacture site, after the PPE 650 has been placed out in the field, or both. The manufacturing process for PPE 650 typically involves embedding within the PPE sufficient software that will allow the device to be more completely initialized at a later time. This manufacturing process may include, for example, testing the bootstrap loader and challenge-response software permanently stored within PPE 650, and loading the PPE's unique ID. These steps provide a basic VDE-capable PPE 650 that may be further initialized (e.g., after it has been installed within an electronic appliance 600 and placed in the field). In some cases, the manufacturing and further initialization process may be combined to produce 'VDE ready' PPEs 650." ('193 223:30-44) 16. "In one example, a person with a laptop 5102 or other computer lacking a PPE 650 wishes nonetheless to take advantage of a subset of secure item delivery services." ('683 62:17-20) 17. "Claims 7-11, ... 99-111 ... are rejected under 35 U.S.C. 103(a) as being unpatentable over Fischer (5,412,717) in view of Narasimhalu et al (5,499,298). Fischer discloses a method and apparatus including a system monitor which

	Claim Term/Phrase	Evidence Supporting MS Construction
		limits the ability of a program about to be executed to the use of predefined resources, ... The set of authorities and restrictions are referred to as 'program authorization information' or 'PAI'. ... A comparison of independent claim 7 to Fischer to derive the similarities and differences between the claimed invention and the prior art follows. ... memory containing a first rule corresponds to a first PAI under a first PCB... Here, Fischer provides a secure container in the form of a program, i.e. a governed item, having an associated PAI, i.e. at least one rule associated with the secure container. A protected processing environment ('PPE') protecting at least some information contained in the PPE, see Fischer Terminal A, and including hardware and/or software used for applying said first rule and the secure container in combination to at least in part govern at least one aspect of access to or use of the governed item, see Fischer at Figure 5 and column 10, lines 8-39 where the first rule in memory is first PCB providing a first PAI and the secure container is a program associated with a second PCB providing a first PAI and the secure container is a program associated with a second PCB having a second PAI associated with the governed item, i.e. the program. ... The difference between claim 7 and Fischer is that the PPE disclosed in Fischer is not explicitly disclosed as protected from tampering by a user of the first apparatus, i.e. terminal A. The Narasimhalu patent ... teaches a method and apparatus for controlling the dissemination of digital information [and] that the end user accesses the digital information with a tamper-proof controlled information access device." (Prosecution History for the 09/221,479 Patent Application, (issued as the '683), Office Action, 11/12/99, pp. 3-5 (IT00065799-801)) 18. "With respect to the remaining issues, Applicants respectfully disagree. For example, the Examiner objects to the use of 'environment' as indefinite and unclear. This word, however, is not used in isolation, but rather in the context of several longer phrases, all of which are defined in the specification. The phrase 'protected processing environment,' for example, is used in Claims 11 and 15-18 and described on at least, for example, pages 7-8 and 25 of the specification. The term 'virtual distribution environment' used in Claim 11 is described, for example, on page 7 of the specification. The terms are also described in the commonly copending application Serial Number 08/388,107 of Ginter et al., filed 13 February 1995, entitled 'System and Methods for Secure Transaction Management and Electronic Rights Protection.' A copy of the incorporated Ginter application can be provided to the Examiner upon request." (Prosecution History for the 08/689,754 Patent Application (issued as the '721), Amendment, 4/14/99, p. 13) (pp. 7, 7-8 and 25 of the original specification are '721 2:62 - 3:13, 2:62 - 3:34 and 8:6-28 of the issued patent) 19. "Another approach to supporting COTS software would use the VDE software running on the user's electronic appliance to create one or more 'virtual machine' environments in which COTS operating system and application programs may run, but from which no information may be permanently stored or otherwise transmitted except under control of VDE." ('193 279:26-40) 20. "VDE may be combined with, or integrated into, many separate computers and/or other electronic appliances. These appliances typically include a secure

	Claim Term/Phrase	Evidence Supporting MS Construction
		subsystem that can enable control of content use such as displaying, encrypting, decrypting, printing, copying, saving, extracting, embedding, distributing, auditing usage, etc. The secure subsystem in the preferred embodiment comprises one or more 'protected processing environments'...." ('193 9:22-29) 21. "The operating system 602 may also support at least one 'application' 608. Generally, 'application' 608 is hardware and/or software specific to the context of appliance 600. For example, if appliance 600 is a personal computer, then 'application' 608 could be a program loaded by the user, for instance, a word processor, a communications system or a sound recorder. If appliance 600 is a television controller box, then application 608 might be hardware or software that allows a user to order videos on demand and perform other functions such as fast forward and rewind. In this example, operating system 602 provides a standardized, well defined, generalized 'interface' that could support and work with many different 'applications' 608." ('193 60:51-64) **Extrinsic:** 1. Processing: "1. The performance of logical operations and calculations on data, including temporary retention of data in processor storage while the data is being operated on." (IBM) 2. Environment: "1. The aggregate of external circumstances, conditions, and objects that affect the development, operation, and maintenance of a system. 2. In computer security, those factors, both internal and external, of an ADP system that help to define the risks associated with its operation." (Longley) 3. "The InterTrust architecture employs three principal components: ... The InterRights Point software provides 'Protected Processing Environment™' technology for manipulating information in DigiBox containers and for securely implementing business rules." (Panel: The InterTrust Commerce Architecture, D. Van Wie et al., 20th NISSC, p. 2, 1997) 4. Environment: See InterTrust node: "A computer that is enabled for processing of DigiBox containers by installation of a PPE, which may be either hardware or software based. A node may include application software and/or operating system integration. The node is also termed the *environment*." (IT Glossary, 8/21/95, TD00068B, IT00032375) 5. Protected Processing Environment (PPE) technology: "The InterTrust technology that provides the protected software environment within the InterRights Point. Protected Processing Environment technology is responsible for the encryption/decryption of data, protected processing of DigiBox containers, and other secure operations, such as protected database access." (IT Glossary, 1997-1998, ML00012B) 6. Protected Processing Environment (PPE): "The PPE is the secure part of a VDE node: either a hardware or software-protected environment in which VDE mechanisms run without external interference. There are various PPE realizations (e.g., physically protected hardware) appropriate to different operational requirements" (IT Glossary, 3/7/95, IT00709619) 7. Secure Processing Unit: "The physically secure hardware component of the SPE: a processor with local memory and non-volatile storage. The SPE consists of the

	Claim Term/Phrase	Evidence Supporting MS Construction
		SPU itself and the SPE software running on the SPU." (IT Glossary, 3/7/95, IT00709620) 8. Protected Processing Environment (PPE): "An InterTrust *node* has a unique *node ID* and contains a *Protected Processing Environment (PPE)* which performs *operations* on *containers* and *control structures* under rules specified by *PERCs* and which may be realized in a tamper resistant hardware component or in tamper-resistant software and a *protected database*, which stores *control objects* and *InterTrust applications*, operating outside the *PPE*, which manipulate *content* and *control objects* through requests to the *PPE*" (IT Glossary, 4/6/95, IT00028206) 9. "All the terms in italics have specific definitions (in the glossary) with respect to InterTrust." 10. "*Global replace of 'VDE' with 'InterTrust' to match new terminology.*" (IT Glossary, 4/6/95, IT00028206) 11. Protected Environment: "A portion of the node software that uses, and protects, the protected node data such as cryptographic keys. The protected environment is responsible for performing all the protected functions for manipulating containers and content; that is, all the operations governed by controls." (IT Glossary, 5/12/95, IT00028294) 12. Protected Processing Environment: (alternate definition): "The protected environment in which the cryptographic and control functions of InterTrust run. The PPE may be protected environmentally (e.g., as a physically protected server machine) or may employ software-based tamper resistance techniques." (IT Glossary, 8/21/95, TD00068B, IT00032377) 13. Secure Processing Environment (SPE): "A hardware-supported realization of the PPE, protected from tampering by physical security techniques. No longer preferred because of the potential confusion between the 'S' in the acronym and 'S' as in 'Software' (which this isn't).. [REPLACEMENT UNCERTAIN]" (IT Glossary, "Obsolete Terminology Section," 5/12/95, IT00028302) 14. Protected Processing Environment (PPE): "The InterTrust protected software environment within the InterTrust Commerce Node. The PPE is responsible for the encryption/decryption of data, protected processing of DigiBox containers, and other secure operations, such as database access." (IT Glossary, 11/17/96, TD00189J, IT00035871) 15. Process: "(1) in computing, the active system entity through which programs run. The entity in a computer system to which authorizations are granted; thus the unit of accountability in a computer system. (2). In computing, a program in execution. ... (4) In computing, a program is a static piece of code and a process is the execution of that code." (Longley)
19.	secure, securely 193.1, 193.11, 193.15 683.2 721.34	**Intrinsic:** Because this term is indefinite and used inconsistently, each use of "secure" and forms thereof in the asserted patents is relevant and herein included by reference. The following examples are illustrative. 1. "HPEs 655 may be provided in two types: secure and not secure." ('193 80:8-9) 2. "Because secondary storage 652 is not secure, SPE 503 must encrypt and cryptographically seal (e.g., using a one-way hash function initialized with a

	Claim Term/Phrase	Evidence Supporting MS Construction
	861.58 891.1 912.8, 912.35	secret value known only inside the SPU 500) each swap block before it writes it to secondary storage." ('193 107:39-42) 3. "Insecure external memory may reduce the wait time for swapped pages to be loaded into SPU 500, but will still incur substantial encryption/decryption penalty for each page." ('193 125:56-59) 4. "The following is a non-exhaustive list of some of the advantageous features provided by ROS 602 in the preferred embodiment: ... Secure secure communications . secure control functions secure virtual memory management information control structures protected from exposure data elements are validated, correlated and access controlled components are encrypted and validated independently components are tightly correlated to prevent unauthorized use of elements control structures and secured executables are validated prior to use to protect against tampering integrates security considerations at the I/O level provides on-the-fly decryption of information at release time enables a secure commercial transaction network flexible key management features" ('193 72:52 - 73:38) 5. "ROS 602 generates component assemblies 690 in a secure matter. As shown graphically in FIGS. 11I and 11J, the different elements comprising a component assembly 690 may be 'interlocking' in the sense that they can only go together in ways that are intended by the VDE participants who created the elements and/or specified the component assemblies. ROS 602 includes security protections that can prevent an unauthorized person from modifying elements, and also prevent an unauthorized person from substituting elements." ('193 84:60 - 85:2) 6. "Because of VDE security, including use of effective encryption, authentication, digital signature, and secure database structures, the records contain within a VDE card arrangement may be accepted as valid transaction records for government and/or corporate recordkeeping requirements." ('193 41:37-42) 7. "In order to maintain security, SPE 503 must encrypt and cryptographically seal each block being swapped out to a storage device external to a supporting SPU 500, and must similarly decrypt, verify the cryptographic seal for, and validate each block as it swapped into SPU 500." ('193 125:60-64) 8. "As mentioned above, memory external to SPU 500 may not be secure. Therefore, when security is required, SPU 500 must encrypt secure information before writing it to external memory before using it." ('193 71:32-36) 9. "Only those processes that execute completely within SPEs 503 (and in some cases, HPEs 655) may be considered to be truly secure. Memory and other resources external to SPE 503 and HPEs 655 used to store and/or process code and/or data to be used in secure processes should only receive and handle that information in encrypted form unless SPE 503/HPE 655 can protect secure process code and/or data form non-secure processes." ('193 81:12-19)

	Claim Term/Phrase	Evidence Supporting MS Construction
		10. "From time to time, two parties (e.g., PPEs A and B), will need to establish a communication channel that is know by both parties to be secure form eavesdropping, secure from tampering, and to be in use solely by the two parties whose identifies are correctly known to each other." ('193 218:33-37) 11. "Since all secure communications are at least in part encrypted and the processing inside the secure subsystem is concealed form outside observation and interference, the present invention ensures that content control information can be enforced." ('193 46:4-8) 12. "VDE 100 provided by the preferred embodiment has sufficient security to help ensure that it cannot be compromised short of a successful 'brute force attack,' and so that the time and cost to succeed in such a 'brute force attack' substantially exceeds any value to be derived. In addition, the security provided by VDE 100 compartmentalizes the internal workings of VDE so that a successful 'brute force attack' would compromise only a strictly bounded subset of protected information, not the entire system." ('193 199:38-47) 13. "Integrity of VDE Security: There are many ways in which a PPE 650 might be compromised. The goal of the security provided by VDE 100 is to reduce the possibility that the system will be compromised, and minimize the adverse effects if it is compromised. The basic cryptographic algorithm that are used to implement VDE 100 are assumed to be safe (cryptographically strong). These include the secret-key encryption of content, public-key signatures for integrity verification, public-key encryption for privacy between PPEs 650 or between a PPE and a VDE administrator, etc. Direct attack on these algorithms is assumed to be beyond the capabilities of an attacker. For domestic versions of VDE 100 some of this probably a safe assumption since the basic building blocks for control information have sufficiently long keys and are sufficiently proven. The following risks of threat or attacks may be significant: Unauthorized creation or modification of component assemblies (e.g., budgets); Unauthorized bulk disclosure of content; Compromise of one or more keys" ('193 221:1-21) 14. See also prior art referenced in the relevant file histories, e.g., Stefik; Tygar et al., "Dyad: A System for Using Physically Secure Coprocessors," School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 (May 1991). 15. "VDE can: (a) audit and analyze the use of content, (b) ensure that content is used only in authorized ways, and (c) allow information regarding content usage to be used only in ways approved by content users." ('193 4:51-56) 16. "Even if the object is stored locally to the VDE node, it may be stored as a secure or protected object so that it is not directly accessible to a calling process." ('193 192:14-17) 17. "An attacker would gain little benefit from intercepting this information since it is transmitted in protected form; she would have to compromise electronic appliance 600(1) or 600(N) (or the SPU 500(1), 500(N)) in order to access this information in unprotected form." ('193 228:25-30) 18. "VDE is a secure system for regulating electronic conduct and commerce. Regulation is ensured by control information put in place by one or more parties." ('193 6:33-35)

	Claim Term/Phrase	Evidence Supporting MS Construction
		19. "It also employs a software object architecture for VDE content containers that carries protected content and may also carry both freely available information (e.g, summary, table of contents) and secured content control information which ensures the performance of control information." ('193 15:41-46) 20. "Because of the breadth of issues resolved by the present invention, it can provide the emerging 'electronic highway' with a single transaction/distribution control system that can, for a very broad range of commercial and data security models, ensure against unauthorized use of confidential and/or proprietary information and commercial electronic transactions." ('193 17:22-28) 21. "VDE can satisfy the requirements of widely differing electronic commerce and data security applications by, in part, employing this general purpose transaction management foundation to securely process VDE transaction related control methods." ('193 25:52-57) 22. "HPE(s) and SPE(s) ... may each perform secure processing based on one or more VDE component assemblies 690, and they may each offer secure processing services to OS kernel 680." ('193 79:41-46) 23. "VDE methods 1000 are designed to provide a very flexible and highly modular approach to secure processing." ('193 181:18-19) 24. "In these cases, secure processing steps performed by an SPU typically must be segmented into small, securely packaged elements that may be 'paged in' and 'paged out' of the limited available internal memory space." (69:43-47) 25. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... VDE employs special purpose hardware distributed throughout some or all locations of a VDE implementation: a) said hardware controlling important elements of: content preparation (such as causing such content to be placed in a VDE content container and associating content control information with said content), content and/or electronic appliance usage auditing, content usage analysis, as well as content usage control; and b) said hardware having been designed to securely handle processing load module control activities, wherein said control processing activities may involve a sequence of required control factors" ('193 21:43 - 22:31) 26. "Memory manager 578 and virtual memory manager 580 in the preferred embodiment manage ROM 532 and RAM 534 memory within SPU 500 in the preferred embodiment. Virtual memory manager 580 provides a fully 'virtual' memory system to increase the amount of 'virtual' RAM available in the SPE secure execution space beyond the amount of physical RAM 534a provided by SPU 500. Memory manager 578 manages the memory in the secure execution space, controlling how it is accessed, allocated and deallocated. SPU MMU 540, if present, supports virtual memory manager 580 and memory manager 578 in the preferred embodiment. In some 'minimal' configurations of SPU 500 there may be no virtual memory capability and all memory management functions will be handled by memory manager 578. Memory management can also be used to help enforce the security provided by SPE 503. In some classes of SPUs 500, for example, the kernel memory manager 578 may use hardware memory management unit (MMU) 540 to provide page level protection within the SPU 500. Such a hardware-based memory management system provides an effective

	Claim Term/Phrase	Evidence Supporting MS Construction
		mechanism for protecting VDE component assemblies 690 from compromise by 'rogue' load modules." ('193 109:24-45) 27. "When a method core 1000' references a load module 1100, a load module is loaded into the SPE 503, decrypted, and then either passed to the electronic appliance microprocessor for executing in an HPE 655 (if that is where it executes), or kept in the SPE (if that is where it executes)." ('193 139:28-31) 28. " The role of go-between 4700 may, in some circumstances, be played by one of the participant's SPU's 500 (PPEs), since SPU (PPE) behavior is not under the user's control, but rather can be under the control of rules and controls provided by one or more other parties other than the user (although in many instances the user can contribute his or her own controls to operate in combination with controls contributed by other parties)." ('683 24:26-33) 29. "Load modules are not necessarily directly governed by PERCs 808 that control them, nor must they contain any time/date information or expiration dates. The only control consideration is the preferred embodiment is that one or more methods 1000 reference them using a correlation tag (the value of a protected object created by the load module's owner, distributed to authorized parties for inclusion in their methods, and to which access and use is controlled by one or more PERCs 808). If a method core 1000' references a load module 1100 and asserts the proper correlation tag (and the load module satisfies the internal tamper checks for the SPE 503), then the load module can be loaded and executed, or it can be acquired from, shipped to, updated, or deleted by, other systems." ('193 139:60 - 140:6) 30. "ROS 602 also provides a tagging and sequencing scheme that may be used within loadable component assemblies 690to detect tampering by substitution. Each element comprising a component assembly 690 may be loaded into a SPU 500, decrypted using encrypt/decrypt engine 522, and then tested/compared to ensure that the proper element has been loaded. ...In addition, a validation/correlation tag stored under the encrypted layer of the loadable element may be compared to make sure it matches on or more tags provided by a requesting process. This prevents unauthorized use of information. As a third protection, a device assigned tag (e.g., a sequence number) stored under an encryption layer of loadable element may be checked to make sure it matches a corresponding tag value expected by SPU 500. This prevents substitution of older elements. Validation/correlation tags are typically passed only in secure wrappers to prevent plaintext exposure of this information outside of SPU 500." ('193 87:41-62) 31. "Key and Tag Manager 558 also provides service relating to tag generation and management. In the preferred embodiment, transaction and access tags are preferably stored by SPE 503 (HPE 665) in protected memory (e.g., within the NVRAM 534b of SPU 500). These tags may be generated by key and tag manager 558. They are used to, for example, check access rights to, validate and correlate data elements. For example, they may be used to ensure components of the secured data structures are not tampered with outside of the SPU 500." ('193 120:59 - 121:1) 32. "Initiation of load module execution in this environment is strictly controlled by a

	Claim Term/Phrase	Evidence Supporting MS Construction
		combination of access tags, validation tags, encryption keys, digital signatures, and/or correlation tags. Thus, a load module 1100 may only be referenced if the caller knows it ID and asserts the shared secret correlation tag specific to that load module. The decrypting SPU may match the identification token an and local access tag of a load module after decryption. These techniques make the physical replacement of any load module 1100 detectable at the next physical access of a load module." ('193 139:45-55) 33. "Meters and budgets are common examples of this. Expiration dates cannot be used effectively to prevent substitution of the previous copy of a budget UDE 1200. To secure these frequently updated items, a transaction tag is generated and included in the encrypted item each time that item is updated. A list of all VDE items Ids and the current transaction tags for each item is maintained as part of the secure database 610." ('193 143:13-20) 34. "UDEs 1200 are preferably encrypted using a site specific key once they are loaded into a site. This site-specific key marks a validation tag that may be derived from a cryptographically strong pseudo-random sequence by the SPE 503 and updated each time the record is written back to the secure database 610. This technique provided reasonable assurance that the UDE 1200 has not been tampered with nor submitted when it is requested by the system for the next use." ('193 143: 29-37) **Extrinsic:** 1. "No data system can be made secure without physical protection of some part of the equipment." (Davies, p. 3) 2. "Security is a negative attribute. We judge a system to be secure if we have not been able to design a method of misusing it which gives some advantage to the attacker." (Davies, p. 4) 3. "Various criteria exist for secure systems - U.S. Dept. of Defense Trusted Computer Security Evaluation Criteria (TCSEC), the Orange Book, Red Book, European and Canadian guidelines, U.S. National Institute of Standards and Technology, and United Kingdom guidelines." (Neumann, p. 233) 4. Security: "1. Protection against unwanted behavior. In present usage, computer security includes properties such as confidentiality, integrity, availability, prevention of denial of service and prevention of generalized misuse. 2. The property that a particular security policy is enforced, with some degree of assurance. 3. Security is sometimes used in the restricted sense of confidentiality, particularly in the case of multilevel security (that is, multilevel confidentiality)." Multilevel Security: "A confidentiality policy based on the relative ordering of multilevel security labels (really multilevel confidentiality, ex. - no adverse flow of information with respect to sensitivity of information)" (Neumann, Glossary and p. 225) 5. "There are two principal objectives: secrecy (or privacy), to prevent unauthorized disclosure of data; and authenticity or integrity) [sic], to prevent the unauthorized modification of data.... Note, however, that whereas it can be used to detect message modification, it cannot prevent it. Encryption alone does not protect against replay, because an opponent could simply replay previous ciphertext."

	Claim Term/Phrase	Evidence Supporting MS Construction
		(Denning, p. 5) 6. "A cipher is unconditionally secure if, no matter how much ciphertext is intercepted, there is not enough information in the ciphertext to determine the plaintext uniquely." (Denning, p. 5) (Davies, pp. 41, 380) 7. "A cipher is computationally secure, or strong, if it cannot be broken by systematic analysis with available resources." (Denning, p. 5) (Davies, pp. 41, 370) 8. Security: "The combination of integrity and secrecy, applied to data." (IT Glossary, 5/12/95, IT00028295) 9. Secrecy: "The inability to obtain any information from data." (IT Glossary, 5/12/95, IT00028294) 10. "... security includes concealment, integrity of messages, authentication of one communicating party by the other..." (Neumann, p. 8) 11. "Computer security rests on confidentiality, integrity, and availability. The interpretations of these three aspects vary, as do the contexts in which they arise. Confidentiality is the concealment of information or resources. ... Confidentiality also applies to the existence of data, which is sometimes more revealing than the data itself. ... All mechanisms that enforce confidentiality require supporting services from the system. The assumption is that the security services can rely on the kernel, and other agents, to supply correct data. Thus, assumptions and trust underlie the confidentiality mechanisms. Integrity refers to the trustworthiness of data or resources, and it is usually phrased in terms of preventing improper or unauthorized change. Integrity includes data integrity (the content of the information) and origin integrity (the source of the data, often called authentication). Integrity mechanisms fall into two classes: prevention mechanisms and detection mechanisms. Protection mechanisms seek to maintain the integrity of the data by blocking any unauthorized attempts to change the data or any attempts to change the data in unauthorized ways. Detection mechanisms do not try to prevent violations of integrity; they simply report that the data's integrity in no longer trustworthy." (Bishop, pp. 4-6) 12. "Definition 4-1. A security policy is a statement that partitions the states of the system into a set of authorized, or secure, states and a set of unauthorized, or nonsecure, states. A secure system is a system that starts in an authorized state and cannot enter an unauthorized state." (Bishop, p. 95) 13. "24.5.1 Secure Systems Systems designed with security in mind have auditing mechanisms integrated with the system design and implementation." (Bishop, p. 706) 14. "Computer security is assuring the secrecy, integrity, and availability of components of computing systems. The three principal pieces of a computing system subject attacks are hardware, software, and data. These three pieces, and the communications between them, constitute the basis of computer security vulnerabilities. This chapter has identified four kinds of attacks on computing systems: interruptions, interceptions, modifications, and fabrications. Three principles affect the direction of work in computer security. By the principle of easiest penetration, a computing system penetrator will use whatever means of

	Claim Term/Phrase	Evidence Supporting MS Construction
		attack is the easiest; therefore. All aspects of computing system security need to be considered at once. By principle of timeliness, a system needs to be protected against penetration only long enough so that penetration is of no value to the penetrator. The principle of effectiveness states that controls must be usable and used in order to serve purpose. Controls can be applied at the levels of data, programs, the system, physical devices, communications links, the environment, and personnel. Sometimes several controls are needed to cover a single vulnerability, and sometimes one control addresses several problems at once." (Pfleeger, p. 4) 15. See also InterTrust's Rule 30(b)(6) testimony 16. See also Microsoft PLR 4-2 Exhs. E & F as revised, e.g. Webster's (1947), p. 1540-41; Pfleeger, p. 4-5; Spencer, Personal Computer Dictionary, p. 156; The Computer Glossary, p. 460; McGraw-Hill Dictionary of Scientific and Technical Terms, p. 1788; Practical Unix Security (O'Reilly 1991), p. 11-12; Bishop, Computer Security (2002) p. 3-24, 47; Hoffman, Modern Methods for Computer Security and Privacy, p. 134-35; Mullender, ed., Distributed Systems (Addison Wesley 2nd ed.), p. 367, 420; Landewehr, "Formal Models for Computer Security" (ACM 1981); Merkle, "Protocols for Public Key Cryptosystems" (IEEE 1980); Cooper, Computer & Communication Security, p. 383; Baker, The Computer Security Handbook, p. 273; Computer Security Handbook, p. 389; Matheson et al., Robustness and Security of Digital Watermarks; National Information Systems Security (INFOSEC) Glossary, p. 49-50; Internet Security Glossary (RFC2828); Tanenbaum, Modern Operating Systems (1992), p. 181-82; IN64706-45, IN176319-72, IT735936 (integrity), IT735938-9 IN00862862, IT1678-96, IT39208-26, IT702969-83, IT399877-80 17. "Secure. Pertaining to the control of who can use an object and to the extent to which the object can be used by controlling the authority given to the user."; "Computer Security. 1. Concepts, techniques, technical measures, and administrative measures used to protect the hardware, software and data of an information processing system from deliberate or inadvertent unauthorized acquisition, damage, destruction, disclosure, manipulation, modification or use or loss. 2. Protection resulting from the application of computer security." (IBM) 18. "Security: Freedom from risk or danger. Safety and assurance of safety"; "secure state - a condition in which none of the subjects in a system can access objects in an unauthorized manner..." (Russell, pp. 8-11, 113, 227, 420) 19. "The protection of computer hardware and software from accidental or malicious access, use, modification, destruction, or disclosure." (Booth) 20. "Prevention of or protection against (a) access to information by unauthorized recipients or (b) intentional but unauthorized destruction or alteration of that information." (Dictionary of Computing, p. 406)

	Claim Term/Phrase	Evidence Supporting MS Construction
		21. "The quality or state of being cost-effectively protected from undue losses (e.g. loss of goodwill, monetary loss, loss of ability to continue operations, etc.)" (Longley). 22. Hoffman, Modern Methods for Computer Security & Privacy, p. 134 23. "Protected Location: A memory location that can only be accessed by an authorized user or process."; "Protected domain: A set of access privileges to protected resources." (Dictionary of Computing) 24. Protect: "To prevent unauthorized access to programs or a computer system; to shield against harm." (Webster's) 25. Protection: "(1) (computing systems). See: Storage protection (2) (software). An arrangement for restricting access to or use of a all, or part, of a computer system."; Storage protection: "An arrangement for preventing access to storage for either reading or writing, or both." (Booth) 26. IN00862862 27. Security: "The combination of integrity and secrecy, applied to data." (IT Glossary, 5/12/95, IT00028295) 28. "Secrecy: The inability to obtain any information from data." (IT Glossary, 5/12/95, IT00028294) 29. Processing: "1. The performance of logical operations and calculations on datum including temporary retention of data in processor storage while the data is being operated on." (IBM) 30. Process: "(1) in computing, the active system entity through which programs run. The entity in a computer system to which authorizations are granted; thus the unit of accountability in a computer system. (2) In computing, a program in execution... . (4) In computing, a program is a static piece of code and a process is the execution of that code." (Longley) 31. Processing: "In legislation, as defined by the U.K. Data Protection Act of 1984, pertaining to the amending, augmenting, deleting, or re-arranging of the data or extracting the information constituting the data and, in the case of personal data, processing means performing any of the abovementioned operations by reference to the data subject." (Longley)
20.	secure container 683.2 861.58 912.35	**Intrinsic:** 1. "Anderson [U.S. Patent No. 5,537,526] does not explicitly address a secure container *per se*, but does place documents into containers [Fig. 8 202] and place restriction via links attached to documents ... which can include restrictions ... Such security tools are rightfully attached to a structure encapsulating the document, e.g. its container." (Prosecution History for the 08/805,804 Patent Application (issued as the '861), Office Action, 6/25/98, p. 5 (MSI 27417-25)) 2. "Claims 7-11, ... are rejected under 35 U.S.C. 103(a) as being unpatentable over Fischer (5,412,717) in view of Narasimhalu et al (5,499,298). ... The set of authorities and restrictions are referred to as 'program authorization information' or 'PAI'. ... A comparison of independent claim 7 to Fischer to derive the similarities and differences between the claimed invention and the prior art follows. ... Here, Fischer provides a secure container in the form of a program, i.e. a governed item, having an associated PAI, i.e. at least one rule associated with the secure container." (Prosecution History for the 09/221,479 Patent

	Claim Term/Phrase	Evidence Supporting MS Construction
		Application (issued as the '683), Office Action, 11/12/99, pp. 3-4 (IT00065799-800)) 3. "1. (Amended) A rights management method comprising: (a) receiving an information signal; (b) steganographically decoding the received information signal to recover digital rights management control information packaged within at least one secure digital container; and (c) performing at least one rights management operation based at least in part on the recovered digital rights management control information. ... Remarks ... For example, amended Claims 1, 15 and 22 each recite a digital secure container in combination. Neither Rhoads [U.S. Patent No. 5,636,292], nor any of the other applied references, teaches or suggests the recited combination of features including any digital secure container." (Prosecution History for the 08/689,606 Patent Application filed 8/12/96) (issued as U.S. Patent 5,943,422, incorporating '107), Amendment, 7/2/98, pp. 1-2, 101 (MSI188164-165, MSI188264) 4. Rhoads, U.S. Patent No. 5,636,292: a. "Fully Exact Steganography Prior art steganographic methods currently known to the inventor generally involve fully deterministic or 'exact' prescriptions for passing a message. Another way to say this is that it is a basic assumption that for a given message to be passed correctly in its entirety, the receiver of the information needs to receive the exact digital data file sent by the sender, tolerating no bit errors or 'loss' of data. By definition, 'lossy' compression and decompression on empirical signals defeat such steganographic methods. (Prior art, such as the previously noted Komatsu work, are the exceptions here.) The principles of this invention can also be utilized as an exact form of steganography proper. It is suggested that such exact forms of steganography, whether those of prior art or those of this invention, be combined with the relatively recent art of the 'digital signature' and/or the DSS (digital signature standard) in such a way that a receiver of a given empirical data file can first verify that not one single bit of information has been altered in the received file, and thus verify that the contained exact steganographic message has not been altered." (Rhoads 55:5-26) b. "One exemplary application is placement of identification recognition units directly within modestly priced home audio and video instrumentation (such as a TV). Such recognition units would typically monitor 'audio and/or video looking for these copyright identification codes, and thence triggering simple decisions based on the findings, such as disabling or enabling recording capabilities, or incrementing program specific billing meters which are transmitted back to a central audio/video service provider and placed onto monthly invoices." (Rhoads 29:23-33) 5. "Use of secure electronic containers to transport items provides an unprecedented degree of security, trustedness and flexibility." ('683 8:50-52) 6. "Even if the object is stored locally to the VDE node, it may be stored as a secure or protected object so that it is not directly accessible to a calling process. ACCESS method 2000 establishes the connections, routings, and security

	Claim Term/Phrase	Evidence Supporting MS Construction
		requisites needed to access the object." ('193 192:14-19) 7. "Electronic delivery person 4060 receives item 4054 in digital form and places it into a secure electronic container 302--thus forming a digital 'object' 300. A digital object 300 may in this case be, for example, as shown in FIGS. 5A and 5B, and may include one or more containers 302 containing item 4054. FIG. 88 illustrates secure electronic container 302 as an attaché case handcuffed to the secure delivery person's wrist. Once again, container is shown as a physical thing for purposes of illustration only--in the example it is preferably electronic rather than physical, and comprises digital information having a well-defined structure (see FIG. 5A). Special mathematical techniques known as 'cryptography' can be used to make electronic container 302 secure so that only intended recipient 4056 can open the container and access the electronic document (or other item) 4054 it contains." ('683 15:56 - 16:6) 8. "[C]ontainer 152 can only be opened within a secure protected processing environment 154 that is part of the virtual distribution environment described in the above-referenced Ginter et al. patent disclosure ...." ('712 168:22-25) 9. "A VDE content container is an object that contains both content (for example, commercially distributed electronic information products such as computer software programs, movies, electronic publications or reference materials, etc.) and certain control information related to the use of the object's content." ('193 19:15-21) 10. "Other applications, such as application 608b shown in FIG. 11B, may not be 'VDE Aware' and therefore may not 'know' how to directly access an interface to VDE functions 604 provided by API 682. To provide for this, ROS 602 may include a 'redirector' 684 that allows such 'non- VDE aware' applications 608(b) to access VDE objects 300 and functions 604. Redirector 684, in the preferred embodiment, translates OS calls directed to the 'other OS functions' 606 into calls to the 'VDE functions' 604. As one simple example, redirector 684 may intercept a 'file open' call from application 608(b), determine whether the file to be opened is contained within a VDE container 300, and if it is, generate appropriate VDE function call(s) to file system 687 to open the VDE container (and potentially generate events to HPE 655 and/or SPE 503 to determine the name(s) of file(s) that may be stored in a VDE object 300, establish a control structure associated with a VDE object 300, perform a registration for a VDE object 300, etc.). Without redirector 684 in this example, a non-VDE aware application such as 608b could access only the part of API 682 that provides an interface to other OS functions 606, and therefore could not access any VDE functions." ('193 82:24-45) 11. "ACCESS method 2000 reads the ACCESS method MDE from the secure database, reads it in accordance with the ACCESS method DTD, and loads encrypted content source and routing information based on the MDE (blocks 2010, 2012). This source and routing information specifies the location of the encrypted content. ACCESS method 2000 then determines whether a connection to the content is available (decision block 2014). This 'connection' could be, for example, an on-line connection to a remote site, a real-time information feed, or a path to a secure/protected resource, for example. If the connection to the content

	Claim Term/Phrase	Evidence Supporting MS Construction
		is not currently available ('No' exit of decision block 2014), then ACCESS method 2000 takes steps to open the connection (block 2016). If the connection fails (e.g., because the user is not authorized to access a protected secure resource), then the ACCESS method 2000 returns with a failure indication (termination point 2018)." ('193 192:36-52) 12. "Appliance 600B may deliver the digital copy of item 4054 within a container 302 and/or may protect the item with seals, electronic fingerprints, watermarks and/or other visible and/or hidden markings to provide a 'virtual container' or some of the security or other characteristics of a container (for example, the ability to associate electronic controls with the item)." ('683 18:49-56) 13. "Trade-offs between flexibility, ease of use and incompatibility and interoperability can be further complicated when security considerations come into play. To be effective in many electronic commerce applications, electronic container designs should be tamper-resistant and secure. One must assume that any tools widely used to create and/or use containers will fall into the hands of those trying to break or crack open the containers or otherwise use digital information without authorization. Therefore, the container creation and usage tools must themselves be secure in the sense that they must protect certain details about the container design. This additional security requirement can make it even more difficult to make containers easy to use and to provide interoperability." ('861 4:51-64) **Extrinsic:** 1. Container: "VDE objects are represented in a special form called a container. The container is implemented within the VDE as an object-oriented container class. The container class provides a standard method by which applications software may encapsulate and read information stored within the object. Additionally, the container may include procedural information associated with the data being stored. Containers may be nested, and share attributes with nested elements. Nested containers are stored within a larger container. VDE recognizes the presence of additional objects within the content, and allows the nested containers to share, extend or override the attributes of an outer container." (VDE ROI DEVICE v1.0a, 2/9/94, IT00008572) 2. Secure: "Pertaining to the control of who can use an object and to the extent to which the object can be used by controlling the authority given to the user." (IBM) 3. Container: "In data security, a multilevel information structure. A container has a classification and may contain objects and/or other containers." (Longley) 4. Container: "A protected (encrypted) storage object that incorporates descriptive information, protected content, and (optionally) control objects applicable to that content." (IT Glossary, 3/7/95, IT00709617) 5. Container: "A contains protected *content*, which is divided into one or more *atomic elements*, and, optionally, *PERCs* governing the *content* and may be manipulated only as specified by a *PERC*." (IT Glossary, 4/6/95, IT00028206) 6. Container: "A packaging mechanism, consisting of: *One or more Element-derived components. *An organization mechanism which provides a unique name

	Claim Term/Phrase	Evidence Supporting MS Construction
		within a flat namespace for each of the components in a Container." (IT Glossary, 5/12/95, IT00028293) 7. Container: "A protected digital information storage and transport mechanism for packaging content and control information." (IT Glossary, 8/21/95, TD00068B, IT00032372) 8. Secure container: "'Secure Container(s)' means electronic container(s) or electronic data arrangements that: (I) use one or more cryptographic or other obfuscation techniques to provide protection for at least a portion of the Content thereof; and (ii) supports the use of Rules and Controls to enable the Management of Content." (License Agreement IT and Universal Music Group, 4/13/99, Exhibit 11 to IT 30(b)(6)) 9. Secure container: "A DigiBox container provides security through encryption and the PPE of a commerce node. A secure container does not require a secure communications transport mode." (IT00035965) 10. "A DigiBox container provides for the persistent protection of its properties." (IT 00035920) 11. "DigiBox containers ensure integrity." (IT00035895)
21.	tamper resistance 721.1	**Intrinsic:** 1. "The level of security and tamper resistance required for trusted SPU hardware processes depends on the commercial requirements of particular markets or market niches, and may vary widely." ('193 49:59-62) **Extrinsic:** 1. Tamper-resistant Module: "In data security, a device in which sensitive information, such as a master cryptographic key, is stored and cryptographic functions are performed. The device has one or more sensors to detect physical attacks, by an adversary trying to gain access to the stored information in which case the stored sensitive data is immediately destroyed." (Longley) 2. See also IT41530-49, IT51147-60 3. "Subversion: A compromise that undermines integrity." (Neumann, p. 349) 4. "Spoofing: Taking on the characteristics of another system or used for purposes of deception. In the present contexts, spoofing is generally prankish rather than overtly malicious, although it is often used elsewhere in a malicious contexts." (Neumann, p. 349) 5. Security: "1. Protection against unwanted behaviors. In present usage, computer security includes properties such as confidentiality, integrity, availability, prevention of denial of service, and prevention of generalized misuse. 2. The property that a particular security policy is enforced, with some degree of assurance. 3. Security is sometimes used in the restricted sense of confidentiality, particularly in the case of multilevel security (that is, multilevel confidentiality)." (Neumann, p. 349)
22.	tamper resistant barrier 721.34	**Intrinsic:** 1. "In addition, Applicants would like to draw the Examiner's attention to other sections of the specification in support of words or phrases cited by the Examiner as 'indefinite.' ... In claims ... 36 ... the term 'barrier' is used as part of the phrase 'tamper resistant barrier.' This phrase is described in the specification on

	Claim Term/Phrase	Evidence Supporting MS Construction
		at least pages 7-8 and 46. In addition, the incorporated Ginter application describes tamper resistant barriers in a number of locations such as, for example, page 201." (Prosecution History for the 08/689,754 Patent Application (issued as the '721), Amendment, 4/14/99, p. 14.) (p. 7 and 46 of the original specification are '721 2:62 - 3:13 and 16:35-54 of the issued patent; p. 201 of Ginter application 08/388,107 is '193 80:40 - 81:1) 2. "SPU 500 is enclosed within and protected by a 'tamper resistant security barrier' 502. Security barrier 502 separates the secure environment 503 from the rest of the world. It prevents information and processes within the secure environment 503 from being observed, interfered with and leaving except under appropriate secure conditions." ('193 59:48-53) 3. "Although block 1262 includes encrypted summary services information on the back up, it preferably does not include SPU device private keys, shared keys, SPU code and other internal security information to prevent this information from ever becoming available to users even in encrypted form." ('193 166:59-64) 4. "Briefly, the preferred example software-based PPE 650 installation process provides the following security techniques: encrypted software distribution, installation customized on a unique instance and/or electronic appliance basis, encrypted on-disk form, installation tied to payment method, unique software and data layout, and identifiable copies." ('900 236:32-42) 5. "... (c) if the load module has an associate digital signature , authenticating the digital signature at least one public key secured behind a tamper resistant barrier and therefore hidden from the user." ('721 22:5-16 (claim 9)) 6. "A further attack technique might involve duplicating one installed operational material 3472 instance by coping the programs and data from one personal computer 3372B to another personal computer 3372C or emulator (see FIG. 67B, block 3364, and the 'copy' arrow 3364A in FIG. 67A). The duplicated PPE instance could be used in a variety of ways, such as, for example, to place an imposter PPE 650 instance on-line and/or to permit further dynamic analysis." ('900 233:8-15) 7. "Various software protection techniques detailed above in connection with FIG. 10 may provide software-based tamper resistant barrier 674 within a software-only and/or hybrid software/hardware protected processing environment 650. The following is an elaboration on those above-described techniques. These software protection techniques may provide, for example, the following: An on-line registration process that results in the creation of a shared secret between the registry and the PPE 650 instance—used by the registry to create content and transactions that are meaningful only to specific PPE instance. An installation program (that may be distinct from the PPE operational material software) that creates a customized installation of the PPE software unique to each PPE instance and/or associate electronic appliance 600. Camouflage protections that make it difficult to reverse engineer the PPE 650 operational materials during PPE 650 operation. Integrity checks performed during PPE 650 operation (e.g., during on-line interactions with trusted servers) to detect compromise. In general, the software-based tamper resistant barrier 674 may establish 'trust' primarily through uniqueness and complexity." ('900 235:30-57)

	Claim Term/Phrase	Evidence Supporting MS Construction
		8. "Operational materials 3472 may then decrypt the next program segment dynamically ... This mechanism increases the tamper-resistant of the executable code-- thus providing additional tamper resistance for PPE operations." ('900 243:3-9) 9. "The software-based tamper resistant barrier 674 provided by HPE 655 may be provided, for example, by: introducing time checks and/or code modifications to complicate the process of stepping through code comprising a portion of kernel 688a and/or a portion of component assemblies 690 using a debugger; using a map of defects on a storage device (e.g., a hard disk, memory card, etc.) to form internal test values to impede moving and/or copying HPE 655 to other electronic appliances 600; using kernel code that contains false branches and other complications in flow of control to disguise internal processes to some degree from disassembly or other efforts to discover details of processes; using 'self-generating' code (based on the output of a co-sine transform, for example) such that detailed and/or complete instruction sequences are not stored explicitly on storage devices and/or in active memory but rather are generated as needed; using code that 'shuffles' memory locations used for data values based on operational parameters to complicate efforts to manipulate such values; using any software and/or hardware memory management resources of electronic appliance 600 to 'protect' the operation of HPE 655 from other processes, functions, etc. Although such a software-based tamper resistant barrier 674 may provide a fair degree of security, it typically will not be as secure as the hardware-based tamper resistant barrier 502 provided (at least in part) by SPU 500." ('193 80:40-65, Fig. 10) 10. "Software-based tamper resistant barrier 674 may be created by software executing on a general-purpose CPU. Various software protection techniques may be used to construct and/or provide software-based tamper resistant barrier 674." ('900 230:61-65) 11. "No software-only tamper resistant barrier 674 can be wholly effective against all of these threats. A sufficiently powerful dynamic analysis (such as one employing an in-circuit emulator) can lay bare all of the software-based PPE 650's secrets. Nonetheless, various techniques described below in connection with FIG. 69A and following make such an analysis extremely frustrating and time consuming--increasing the 'work factor' to a point where it may become commercially unfeasible to attempt to 'crack' a software-based tamper resistant barrier 674." ('900 233:24-33) 12. "For example, the PPE 650 may rewrite or overwrite memory locations immediately after using same to make their contents unavailable for scrutiny. Similarly, the PPE 650 operational software may use hardware and/or time dependent sequences to prevent emulation. Additionally, some of the PPE 650 environment code may be self-modifying." ('900 236:9-15) **Extrinsic:** 1. Tamper-resistant module: "In data security, a device in which sensitive information, such as a master cryptographic key, is stored and cryptographic functions are performed. The device has one or more sensors to detect physical attacks, by an adversary trying to gain access to the stored information in which

	Claim Term/Phrase	Evidence Supporting MS Construction
		case the stored sensitive data is immediately destroyed." (Longley) 2. "The 'tamper-resistant module' is physically strong and destroys secrets when opened, and the software running inside has been checked for integrity;" (Davies, p. 3) 3. "The host computer is provided with a specially, physically secure module containing all the secret information which must be protected. In the IBM papers it is called the 'Cryptographic Facility': we shall call it a 'Tamper Resistant Module' (TRM)." (Davies, p. 144)
23.	use 193.19 683.2 721.1 861.58 891.1 912.8, 912.35	**Intrinsic:** 1. "Provides non-repudiation of use and may record specific forms of use such as viewing, editing, extracting, copying, redistributing (including to what one or more parties), and/or saving." ('683 6:46-48) 2. "Content (executables for example) delivered with proof of delivery and/or execution or other use." ('683 7:8-9) 3. "In general, VDE enables parties that (a) have rights in electronic information, and/or (b) act as direct or indirect agents for parties who have rights in electronic information, to ensure that the moving, accessing, modifying, or otherwise using of information can be securely controlled by rules regarding how, when, where, and by whom such activities can be performed." ('193 6:24-31) 4. "Some or all of the back up files may be packaged within an administrative object and transmitted for analysis, transportation, or other uses." ('193 167:45-48) 5. "to securely control access and other use, including distribution of records, documents, and notes associated with the case." ('193 274:34-36) 6. "Thus wrapped, a VDE object may be distributed to the recipient without fear of unauthorized access and/or other use. The one or more authorized users who have received an object are the only parties who may open that object and view and/or manipulate and/or otherwise modify its contents and VDE secure auditing ensures a record of all such user content activities." ('193 277:15-21) 7. "These appliances typically include a secure subsystem that can enable control of content use such as displaying, encrypting, decrypting, printing, copying, saving, extracting, embedding, distributing, auditing usage, etc." ('193 9:24-27) 8. "VDE provides a secure, distributed electronic transaction management system for controlling the distribution and/or other usage of electronically provided and/or stored information." ('193 9:36-39) 9. "As a result, VDE supports most types of electronic information and/or appliance: usage control (including distribution), security, usage auditing, reporting, other administration, and payment arrangements." ('193 13:50-53) 10. "SPU 500 is enclosed within and protected by a 'tamper resistant security barrier' 502. Security barrier 502 separates the secure environment 503 from the rest of the world. It prevents information and processes within the secure environment 503 from being observed, interfered with and leaving except under appropriate secure conditions. Barrier 502 also controls external access to secure resources, processes and information within SPU 500. In one example, tamper resistant security barrier 502 is formed by security features such as 'encryption,' and hardware that detects tampering and/or destroys sensitive information within secure environment 503 when tampering is detected." ('193 59:48-59)

	Claim Term/Phrase	Evidence Supporting MS Construction
		11. "Once the information is downloaded, the now-initialized PPE 650 can discard (or simply not use) the manufacturing key." ('193 212:57-59) **Extrinsic:** 1. User: "A person using a InterTrust node to perform some function (i.e., acting in some role). A user is identified with respect to the node by a user ID." (IT Glossary, 5/12/95, IT00028300) 2. User ID: "Locally to a InterTrust node, each InterTrust user has an ID associated with a user name and authentication (e.g., password). In some deployments, there may be only one user, and access to the machine may be considered sufficient authentication; in such cases, the user ID concept may not be visible to the user even though it is present in the implementation." (IT Glossary, 5/12/95, IT00028301) 3. Use: "To use an object is to access the content. This involves the processes of controlling and metering the use of the property and creating audit trail records on the use." (VDE ROI DEVICE v1.0a, 2/9/94, IT00008570)
24.	virtual distribution environment 900.155 Also as set forth in each "claim as a whole" by Microsoft.	Virtual Distribution Environment: "CLAIM AS A WHOLE": **Intrinsic:** 1. "The instant application is one of a series of applications which are all generally directed to a virtual distribution environment." (09/208,017 ('193), Examiner's Amendment, 8/4/00, p. 2) 2. See generally Background and Summary of Invention of '193 Patent ('193 2:22 - 49:63) 3. "With respect to the remaining issues, Applicants respectfully disagree. For example, the Examiner objects to the use of 'environment' as indefinite and unclear. This word, however, is not used in isolation, but rather in the context of several longer phrases, all of which are defined in the specification. The phrase 'protected processing environment,' for example, is used in Claims 11 and 15-18 and described on at least, for example, pages 7-8 and 25 of the specification. The term 'virtual distribution environment' used in Claim 11 is described, for example, on page 7 of the specification. The terms are also described in the commonly copending application Serial Number 08/388,107 of Ginter et al., filed 13 February 1995, entitled 'System and Methods for Secure Transaction Management and Electronic Rights Protection.' A copy of the incorporated Ginter application can be provided to the Examiner upon request." 08/689,754 ('721), Amendment, 4/14/99, p. 13 (pp. 7, 7-8 and 25 of the original specification are '721 2:62 - 3:13, 2:62 - 3:34 and 8:6-28 of the issued patent) 4. See also, Prosecution History of '900: Claims 302, 321 and 322, as pending: "302. A virtual distribution environment comprising • a first host processing environment comprising • a central processing unit;

	Claim Term/Phrase	Evidence Supporting MS Construction
		• main memory operatively connected to said central processing unit; • mass storage operatively connected to said central processing unit and said main memory; • said mass storage storing tamper resistant software designed to be loaded into said main memory and executed by said central processing unit, said tamper resistant software comprising: • machine check programming which derives information from one or more aspects of said host processing environment, • one or more storage locations storing said information; and • integrity programming which • causes said machine check programming to derive said information, • compares said information to information previously stored in said one or more storage locations, and • generates an indication based on the result of said comparison. 321. A virtual distribution environment as in claim 302, • said virtual distribution environment further comprising programming which takes one or more actions based on the state of said indication. 322. A virtual distribution environment as in claim 321 in which said one or more actions includes at least temporarily halting further processing." (Prosecution History for Patent Application 08/706,206 (issued as the '900 patent), Amendment, 06/09/98, 92-93, 96, 96-97)) b. "Claims ... 322-324, ... are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims." (Prosecution History for Patent Application 08/706,206, Office Action, 08/27/98, p. 2) c. "322. A virtual distribution environment comprising • a first host processing environment comprising • a central processing unit; • main memory operatively connected to said central processing unit; • mass storage operatively connected to said central processing unit and said main memory; • said mass storage storing tamper resistant software designed to be loaded into said main memory and executed by said central processing unit, said tamper resistant software comprising: • machine check programming which derives information from one or more aspects of said host processing environment, • one or more storage locations storing said information; • integrity programming which o causes said machine check programming to derive said information, o compares said information to information previously stored in said one or more storage locations, and o generates an indication based on the result of said comparison; and • programming which takes one or more actions based on the state of

	Claim Term/Phrase	Evidence Supporting MS Construction
		said indication; • said one or more actions including at least temporarily halting further processing." ... Remarks, "Applicants appreciate the indication that claims ... are allowed and that claims ... 322-324 are objected to but would be allowable if rewritten into independent form. ... For purposes of expedition, applicants are cancelling the rejected claims without prejudice ..., and are rewriting objected to dependent claims into independent form." (Prosecution History for Patent Application 08/706,206, Amendment, 11/23/98, p. 27-28, 42) (1) DATA SECURITY AND COMMERCE WORLD: **Intrinsic:** 1. "VDE supports a model wide, distributed security implementation which creates a single secure 'virtual' transaction processing and information storage environment. VDE enables distributed VDE installations to securely store and communicate information and remotely control the execution processes and the character of use of electronic information at other VDE installations and in a wide variety of ways. . . " ('193 21:57-65) 2. "The rights protection problems solved by the present invention are electronic versions of basic societal issues. These issues include protecting property rights, protecting privacy rights, properly compensating people and organizations for their work and risk, protecting money and credit, and generally protecting the security of information." ('193 4:8-13) 3. "The present invention provides a new kind of 'virtual distribution environment' (called 'VDE' in this document) that secures, administers, and audits electronic information use. VDE also features fundamentally important capabilities for managing content that travels 'across' the 'information highway.'" ('193 2:24-28) 4. "A fundamental problem for electronic content providers is extending their ability to control the use of proprietary information. Content providers often need to limit use to authorized activities and amounts. Participants in a business model involving, for example, provision of movies and advertising on optical discs may include actors, directors, script and other writers, musicians, studios, publishers, distributors, retailers, advertisers, credit card services, and content end-users. These participants need the ability to embody their range of agreements and requirements, including use limitations, into an 'extended' agreement comprising an overall electronic business model. This extended agreement is represented by electronic content control information that can automatically enforce agreed upon rights and obligations. Under VDE, such an extended agreement may comprise an electronic contract involving all business model participants. Such an agreement may alternatively, or in addition, be made up of electronic agreements between subsets of the business model participants. Through the use of VDE, electronic commerce can function in the same way as traditional commerce-that

	Claim Term/Phrase	Evidence Supporting MS Construction
		is commercial relationships regarding products and services can be shaped through the negotiation of one or more agreements between a variety of parties." ('193 2:37-60) 5. "Protecting the rights of electronic community members involves a broad range of technologies. VDE combines these technologies in a way that creates a 'distributed' electronic rights protection 'environment.' This environment secures and protects transactions and other processes important for rights protection. VDE, for example, provides the ability to prevent, or impede, interference with and/or observation of, important rights related transactions and processes." ('193 3:63 - 4:3) 6. "VDE is a cost-effective and efficient rights protection solution that provides a unified, consistent system for securing and managing transaction processing. VDE can: (a) audit and analyze the use of content, (b) ensure that content is used only in authorized ways, and (c) allow information regarding content usage to be used only in ways approved by content users." ('193 4:48-55) 7. "In general, VDE enables parties that (a) have rights in electronic information, and/or (b) act as direct or indirect agents for parties who have rights in electronic information, to ensure that the moving, accessing, modifying, or otherwise using of information can be securely controlled by rules regarding how, when, where, and by whom such activities can be performed." ('193 6:24-30) 8. "A variety of capabilities are required to implement an electronic commerce environment. VDE is the first system that provides many of these capabilities and therefore solves fundamental problems related to electronic dissemination of information." ('193 8:16-20) 9. "VDE offers an architecture that avoids reflecting specific distribution biases, administrative and control perspectives, and content types. Instead, VDE provides a broad-spectrum, fundamentally configurable and portable, electronic transaction control, distributing, usage, auditing, reporting, and payment operating environment. VDE is not limited to being an application or application specific toolset that covers only a limited subset of electronic interaction activities and participants. Rather, VDE supports systems by which such applications can be created, modified, and/or reused. As a result, the present invention answers pressing, unsolved needs by offering a system that supports a standardized control environment which facilitates interoperability of electronic appliances, interoperability of content containers, and efficient creation of electronic commerce applications and models through the use of a programmable, secure electronic transactions management foundation and reusable and extensible executable components. VDE can support a single electronic 'world' within which most forms of electronic transaction activities can be managed." ('193 8:53 - 9:5) 10. "VDE can securely manage the integration of control information provided by two or more parties. As a result, VDE can construct an electronic agreement between VDE participants that represent a 'negotiation' between, the control requirements of, two or more parties and enacts terms and conditions of a resulting agreement. VDE ensures the rights of each party to an electronic agreement regarding a wide range of electronic activities related to electronic

	Claim Term/Phrase	Evidence Supporting MS Construction
		information and/or appliance usage." ('193 9:52-61) 11. "'Hardware' 506 also contains long-term and short-term memories to store information securely so it can't be tampered with." ('193 60:1-3) 12. "VDE prevents many forms of unauthorized use of electronic information, by controlling and auditing (and other administration of use) electronically stored and/or disseminated information." ('193 11:60-63) 13. "Together, these VDE components comprise a secure, virtual, distributed content and/or appliance control, auditing (and other administration), reporting, and payment environment." ('193 13:14-17) 14. "VDE can securely deliver information from one party to another concerning the use of commercially distributed electronic content. Even if parties are separated by several 'steps' in a chain (pathway) of handling for such content usage information, such information is protected by VDE through encryption and/or other secure processing. Because of that protection, the accuracy of such information is guaranteed by VDE, and the information can be trusted by all parties to whom it is delivered." ('193 14:31-39) 15. "VDE allows the needs of electronic commerce participants to be served and it can bind such participants together in a universe wide, trusted commercial network that can be secure enough to support very large amounts of commerce. VDE's security and metering secure subsystem core will be present at all physical locations where VDE related content is (a) assigned usage related control information (rules and mediating data), and/or (b) used. This core can perform security and auditing functions (including metering) that operate within a 'virtual black box,' a collection of distributed, very secure VDE related hardware instances that are interconnected by secured information exchange (for example, telecommunication) processes and distributed database means." ('193 15:14-27) 16. "VDE provides organization, community, and/or universe wide secure environments whose integrity is assured by processes securely controlled in VDE participant user installations (nodes)." ('193 20:48-51) 17. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention: VDE employs a variety of capabilities that serve as a foundation for a general purpose, sufficiently secure distributed electronic commerce solution. VDE enables an electronic commerce marketplace that supports divergent, competitive business partnerships, agreements, and evolving overall business models. For example, ... employ 'templates' to ease the process of configuring capabilities of the present invention as they relate to specific industries or businesses.... Given the very large range of capabilities and configurations supported by the present invention, reducing the range of configuration opportunities to a manageable subset particularly appropriate for a given business model allows the full configurable power of the present invention to be easily employed by 'typical' users who would be otherwise burdened with complex programming and/or configuration design responsibilities template applications can also help ensure that VDE related processes are secure and optimally bug free by reducing the risks associated with the contribution of independently developed load modules, including unpredictable aspects of code interaction between independent modules and applications, as well as security

	Claim Term/Phrase	Evidence Supporting MS Construction
		risks associated with possible presence of viruses in such modules.... As the context surrounding these templates changes or evolves, template applications provided under the present invention may be modified to meet these changes for broad use, or for more focused activities .... Of course, templates may, under certain circumstances have fixed control information and not provide for user selections or parameter data entry." ('193 21:43-53; 27:1 - 28:18) 18. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention:... provide mechanisms to persistently maintain trusted content usage and reporting control information through both a sufficiently secure chain of handling of content and content control information and through various forms of usage of such content wherein said persistence of control may survive such use. Persistence of control includes the ability to extract information from a VDE container object by creating a new container whose contents are at least in part secured and that contains both the extracted content and at least a portion of the control information which control information of the original container and/or are at least in part produced by control information of the original container for this purpose and/or VDE installation control information stipulates should persist and/or control usage of content in the newly formed container. Such control information can continue to manage usage of container content if the container is 'embedded' into another VDE managed object, such as an object which contains plural embedded VDE containers, each of which contains content derived (extracted) from a different source." ('193 21:43-45; 28:45-65) 19. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... Interoperability is fundamental to efficient electronic commerce. The design of the VDE foundation, VDE load modules, and VDE containers, are important features that enable the VDE node operating environment to be compatible with a very broad range of electronic appliances." ('193 21:43-45; 34:25-30) 20. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... securely support electronic currency and credit usage control, storage, and communication at, and between, VDE installations." ('193 21:43-45; 36:49-51) 21. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... requiring reporting and payment compliance by employing exhaustion of budgets and time ageing of keys." ('193 21:43-45; 40:8-9) 22. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... Because of the VDE security, including use of effective encryption, authentication, digital signaturing, and secure database structures, the records contained within a VDE card arrangement may be accepted as valid transaction records for government and/or corporate recordkeeping requirements." ('193 21:43-45; 41:37-42) 23. "Since all secure communications are at least in part encrypted and the processing inside the secure subsystem is concealed from outside observation and interference, the present invention ensures that content control information can be enforced." ('193 46:4-8)

	Claim Term/Phrase	Evidence Supporting MS Construction
		24. "An important feature of VDE is that it can be used to assure the administration of, and adequacy of security and rights protection for, electronic agreements implemented through the use of the present invention." ('193 46:51-54) 25. "These are merely a few simple examples demonstrating the importance of ROS 602 ensuring that certain component assemblies 690 are formed in a secure manner. ROS 602 provides a wide range of protections against a wide range of 'threats' to the secure handling and execution of component assemblies 690." ('193 85:15-20) 26. "VDE further enables this process by providing a secure execution space in which the negotiation process(es) are assured of integrity and confidentiality in their operation." ('193 245:20-22) 27. "Taken together, and employed at times with VDE administrative objects and VDE security arrangements and processes, the present invention truly achieves a content control and auditing architecture that can be configured to most any commercial distribution embodiment." ('193 261:10-15) 28. "For example, VDE 100 positively controls content access and usage, provides guarantee of payment for content used, and enforces budget limits for accessed content." ('193 240:53-56) 29. "Such metering is a flexible basis for ensuring payment for content royalties, licensing, purchasing, and/or advertising." ('193 33:56-58) 30. "The overall integrity and security of VDE 100 could ensure, in a coherent and centralized manner, that electronic reporting of tax related information (derived from one or more electronic commerce activities) would be valid and comprehensive." ('193 237:47-51) 31. "Distributors 106 and financial clearinghouses 116 may themselves be audited based on secure records of their administrative activities and a chain of reliable, 'trusted' processes ensures the integrity of the overall digital distribution process. This allows content owners, for example, to verify that they are receiving appropriate compensation based on actual content usage or other agreed-upon bases." ('193 254:66 - 255:5) 32. "Because the control information is carried with each copy of a VDE protected document, and can ensure that central registries are updated and/or that originators are notified of document use, tracking can be prompt and accurate." ('193 281:14-16) 33. "A final desirable feature of agreements in general (and electronic representations of agreements in particular) is that they be accurately recorded in a non-repudiatable form. In traditional terms, this involves creating a paper document (a contract) that describes the rights, restrictions, and obligations of all parties involved. This document is read and then signed by all parties as being an accurate representation of the agreement. Electronic agreements, by their nature, may not be initially rendered in paper. VDE enables such agreements to be accurately electronically described and then electronically signed to prevent repudiation." ('193 245:25-35) 34. "As discussed above, a wide variety of techniques are currently being used to provide secure, trusted confidential delivery of documents and other items. Unfortunately, none of these previously existing mechanisms provide truly

	Claim Term/Phrase	Evidence Supporting MS Construction
		trusted, virtually instantaneous delivery on a cost-effective, convenient basis and none provide rights management and auditing through persistent, secure, digital information protection. In contrast, the present inventions provide the trustedness, confidentiality and security of a personal trusted courier on a virtually instantaneous and highly cost-effective basis. They provide techniques, systems and methods that can being to any form of electronic communications (including, but not limited to Internet and internal company electronic mail) an extremely high degree of trustedness, confidence and security approaching or exceeding that provided by a trusted personal courier. They also provide a wide variety of benefits that flow from rights management and secure chain of handling and control." ('683 5:22-40) 35. "The Virtual Distribution Environment provides comprehensive overall systems, and wide arrays of methods, techniques, structures and arrangements, that enable secure, efficient electronic commerce and rights management on the Internet and other information superhighways and on internal corporate networks such as 'Intranets'." ('683 5:41-51-56) 36. "Parties using the Virtual Distribution Environment can participate in commerce and other transactions in accordance with a persistent set of rules they electronically define." ('683 6:11-14) 37. "All of these various coordination steps can be performed nearly simultaneously, efficiently, rapidly and with an extremely high degree of trustedness based on the user of electronic containers 302 and the secure communications, authentication, notarization and archiving techniques provided in accordance with the present inventions." ('683 55:54-59) 38. "People are increasingly using secure digital containers to safely and securely store and transport digital content. One secure digital container model is the 'DigiBox™' container developed by InterTrust Technologies, Inc. of Sunnyvale, Calif. The Ginter et al. patent specification referenced above describes many characteristics of this DigiBox™ container model—a powerful, flexible, general construct that enables protected, efficient and interoperable electronic description and regulation of electronic commerce relationship of all kinds, including the secure transport, storage and rights management interface with objects and digital information within such containers." ('861 1:35-41) 39. "Briefly, DigiBox containers are tamper-resistant digital containers that can be used to package any kind of digital information such as, for example, text, graphics, executable software, audio and/or video. The rights management environment in which DigiBox™. containers are used allows commerce participants to associate rules with the digital information (content). The rights management environment also allows rules (herein including rules and parameter data controls) to be securely associated with other rights management information, such as for example, rules, audit records created during use of digital information and administrative information associated with keeping the environment working properly, including ensuring rights and any agreements among parties. The DigiBox™.. electronic container can be used to store, transport and provide a rights management interfaces to digital information, related rules and other rights management information, as well as to other objects

	Claim Term/Phrase	Evidence Supporting MS Construction
		and/or data within a distributed, rights management environment. This arrangement can be used to provide electronically enforced chain of handling and control wherein rights management persists as a container moves from one entity to another. This capability helps support a digital rights management architecture that allows content rightsholders (including any parties who have system authorized interests related to such content, such as content republishes or even governmental authorities) to securely control and manage content, events, transactions, rules and usage consequences, including any required payment and/or usage reporting. This secure control and management continues persistently, protecting rights as content is delivered to, used by, and passed among creators, distributors, repurposes, consumers, payment disagregators, and other value chain participants." ('861 1:47 - 2:12) 40. "Use of secure electronic containers to transport items provides an unprecedented degree of security, trustedness and flexibility." ('683 8:50-52) 41. "Virtual distribution environment 100 is 'virtual' because it does not require many of the physical 'things' that used to be necessary to protect rights, ensure reliable and predictable distribution, and ensure proper compensation to content creators and distributors." ('193 53:23-27) **Extrinsic:** 42. VDE: "VDE is the broad name given to a comprehensive system (algorithms, software, and hardware) that provides metering, securing, and administration tools for intellectual property. VDE stands for 'Virtual Distribution Environment.'" (VDE ROI DEVICE v1.0a, 2/9/94, IT00008570) 43. Virtual: "Pertaining to a functional unit that appears to be real, but whose functions are accomplished by other means." (IBM) 44. Environment: "1. The aggregate of external circumstances, conditions, and objects that affect the development, operation, and maintenance of a system. 2. In computer security, those factors, both internal and external, of an ADP system that help to define the risks associated with its operation." (Longley) 45. Environment: See InterTrust node: "A computer that is enabled for processing of DigiBox containers by installation of a PPE, which may be either hardware or software based. A node may include application software and/or operating system integration. The node is also termed the *environment*." (IT Glossary, 8/21/95, TD00068B, IT00032375) 46. InterTrust Commerce Architecture model: "A model that defines a general-purpose distributed architecture for secure electronic commerce and digital rights management. The InterTrust Commerce Architecture model includes four key software elements: DigiBox secure containers, InterRights Point software with associated protected database, the InterTrust Transaction Authority Framework, and the InterTrust Deployment Manager." (IT Glossary, 1997, ML00012A) 47. VDE is a system using secure computing technology to enforce a chain of handling and control representing the rights of interested parties. (IT Glossary, 3/7/95, IT00709616) 48. Virtual Distribution Environment (VDE): "A set of components that protects content and enforces rights associated with content." (IT Glossary, 3/7/95,

	Claim Term/Phrase	Evidence Supporting MS Construction
		IT00709620) 49. "Virtual Distribution Environment: or 'VDE' shall mean a system which guarantees: (I) that the content creators, publishers, and/or distributors of information receive agreed upon fees for the use of, and/or records of the use of, electronic content; and/or (ii) that stored and/or distributed information will be used only in authorized ways. More particularly, VDE relates to systems for applying controls to, and controlling and/or auditing use of, electronically stored and/or disseminated information." (License Agreement, National Semiconductor and EPR, 3/18/94, Exhibit 12 to IT 30(b)(6)) 50. See also IT0001689-96, IT0709785 (VDE on a Page), IT000202-29 (2) SECURE PROCESSING ENVIRONMENT: 1. "VDE allows the needs of electronic commerce participants, to be served and it can bind such participants together in a universe wide, trusted commercial network that can be secure enough to support very large amounts of commerce. VDE's security and metering secure subsystem core will be present all physical locations where VDE related contents is (a) assigned usage related control information (rules and mediating data), and/or (b) used. This core can perform security and auditing functions (including metering) that operate within a 'virtual black box,' a collection of distributed, very secure VDE related hardware instances that are interconnected by secured information exchange (for example, telecommunication) processes and distributed database means." ('193 15:14-27) 2. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... VDE employs special purpose hardware distributed throughout some or all locations of a VDE implementation: a) said hardware controlling important elements of: content preparation (such as causing such content to be placed in a VDE content container and associating content control information with said content), content and/or electronic appliance usage auditing, content usage analysis, as well as content usage control; and b) said hardware having been designed to securely handle processing load module control activities, wherein said control processing activities may involve a sequence of required control factors" ('193 21:43-45; 22:20-31) 3. "Physical facility and user identity authentication security procedures may be used instead of hardware SPUs at certain nodes, such as at an established financial clearinghouse, where such procedures may provide sufficient security for trusted interoperability with a VDE arrangement employing hardware SPUs at user nodes." ('193 45:60-65) 4. "An important part of VDE provided by the present invention is the core secure transaction control arrangement, herein called an SPU (or SPUs), that typically must be present in each user's computer, other electronic appliance, or network. SPUs provide a trusted environment for generating decryption keys, encrypting and decrypting information, managing the secure communication of keys and other information between electronic appliances (i.e. between VDE installations and/or between plural VDE instances within a single VDE installation), securely accumulating and managing audit trail, reporting, and budget

	Claim Term/Phrase	Evidence Supporting MS Construction
		information in secure and/or non-secure non-volatile memory, maintaining a secure database of control information management instructions, and providing a secure environment for performing certain other control and administrative functions." ('193 48:66 - 49:17) 5. "A hardware SPU (rather than a software emulation) within a VDE node is necessary if a highly trusted environment for performing certain VDE activities is required." ('193 49:15-17) 6. "'Hardware' 506 also contains long-term and short-term memories to store information securely so it can't be tampered with." ('193 60:1-3) 7. "A VDE node's hardware SPU is a core component of a VDE secure subsystem and may employ some or all of an electronic appliance's primary control logic, such as a microcontroller, microcomputer or other CPU arrangement. This primary control logic may be otherwise employed for non VDE purposes such as the control of some or all of an electronic appliance's non-VDE functions. When operating in a hardware SPU mode, said primary control logic must be sufficiently secure so as to protect and conceal important VDE processes. For example, a hardware SPU may employ a host electronic appliance microcomputer operating in protected mode while performing VDE related activities, thus allowing portions of VDE processes to execute with a certain degree of security." ('193 49:33-46) 8. "As shown FIG. 6 [sic], in the preferred embodiment, an SPU 500 may be implemented as a single integrated circuit 'chip' 505 to provide a secure processing environment in which confidential and/or commercially valuable information can be safely processed, encrypted and/or decrypted." ('193 63:48-52) 9. "SPU 500 is enclosed within and protected by a 'tamper resistant security barrier' 502. Security barrier 502 separates the secure environment 503 from the rest of the world. It prevents information and processes within the secure environment 503 form being observed, interfered with and leaving except under appropriate secure conditions. Barrier 502 also controls external access to secure resources, processes and information within SPU 500. In one example, tamper resistant security barrier 502 is formed by security features such as 'encryption,' and hardware that detects tampering and/or destroys sensitive information within secure environment 503 when tampering is detected." ('193 59:48-59) 10. "SPU 500 may be surrounded by a tamper-resistant hardware security barrier 502. Part of this security barrier 502 is formed by a plastic or other package in which an SPU 'die' is encased. Because the processing occurring within, and information stored by, SPU 500 are not easily accessible to the outside world, they are relatively secure from unauthorized access and tampering. All signals cross barrier 502 through a secure, controlled path provided by BIU 530 that restricts the outside world's access to the internal components within SPU 500. The secure, controlled path resists attempts form the outside world to access secret information and resources within SPU 500." ('193 63:60 - 64:5)

	Claim Term/Phrase	Evidence Supporting MS Construction
		(3) VDE CONTROLS: See support as listed for Control (n.) , item #8, above. 1. "Limited only by the VDE control information employed by content creators, other providers, and other pathway of handling and control participants, VDE allows a 'natural' and unhindered flow of, and creation of, electronic content product models." ('193 297:25-29) 2. "Regulation is ensured by control information put in place by one or more parties." ('193 6:34-35) 3. "As a result, the present invention answers pressing, unsolved needs by offering a system that supports a standardized control environment which facilitates interoperability of electronic appliances, interoperability of content containers, and efficient creation of electronic commerce applications and models through the use of a programmable, secure electronic transactions management foundation and reusable and extensible executable components." ('193 8:62 - 9:3) 4. "Independently, securely deliverable, component based control information allows efficient interaction among control information sets supplied by different parties." ('193 10:46-50) 5. "A significant facet of the present invention's ability to broadly support electronic commerce is its ability to securely manage independently delivered VDE component objects containing control information (normally in the form of VDE objects containing one or more methods, data, or load module VDE components). This independently delivered control information can be integrated with senior and other pre-existing content control information to securely form derived control information using the negotiation mechanisms of the present invention. All requirements specified by this derived control information must be satisfied before VDE controlled content can be accessed or otherwise used. This means that, for example, all load modules and any mediating data which are listed by the derived control information as required must be available and securely perform their required function." ('193 10:66 - 11:14) 6. "Content control information governs content usage according to criteria set by holders of rights to an object's contents and/or according to parties who otherwise have rights associated with distributing such content (such as governments, financial credit providers, and users)." ('193 15:46-48) 7. "In part, security is enhanced by object methods employed by the present invention because the encryption schemes used to protect an object can efficiently be further used to protect the associated content control information (software control information and relevant data) from modification." ('193 15:51-55) 8. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... Content users, such as end-user customers using commercially distributed content (games, information resources, software programs, etc.), can define, if allowed by senior control information, budgets, and/or other control information, to manage their own internal use of content." ('193 21:43-45; 29:3-8) 9. "Summary of Some Important Features Provided by VDE in Accordance With

	Claim Term/Phrase	Evidence Supporting MS Construction
		the Present Invention.... support the separation of fundamental transaction control processes through the use of event (triggered) based method control mechanisms. These event methods trigger one or more other VDE methods (which are available to a secure VDE sub-system) and are used to carry out VDE managed transaction related processing. These triggered methods include independently (separably) and securely processable component billing management methods, budgeting management methods, metering management methods, and related auditing management processes. As a result of this feature of the present invention, independent triggering of metering, auditing, billing, and budgeting methods, the present invention is able to efficiently, concurrently support multiple financial currencies (e.g. dollars, marks, yen) and content related budgets, and/or billing increments as well as very flexible content distribution models." ('193 21:43-45; 42:21-38) 10. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention....support, complete, modular separation of the control structures related to (1) content event triggering, (2) auditing, (3) budgeting (including specifying no right of use or unlimited right of use), (4) billing, and (5) user identity (VDE installation, client name, department, network, and/or user, etc.). The independence of these VDE control structures provides a flexible system which allows plural relationships between two or more of these structures, for example, the ability to associate a financial budget with different event trigger structures (that are put in place to enable controlling content based on its logical portions). Without such separation between these basic VDE capabilities, it would be more difficult to efficiently maintain separate metering, budgeting, identification, and/or billing activities which involve the same, differing (including overlapping), or entirely different, portions of content for metering, billing, budgeting, and user identification, for example, paying fees associated with usage of content, performing home banking, managing advertising services, etc. VDE modular separation of these basic capabilities supports the programming of plural, 'arbitrary' relationships between one or differing content portions (and/or portion units) and budgeting, auditing, and/or billing control information." ('193 21:43-45; 42:39-63) 11. "The virtual distribution environment 100 prevents use of protected information except as permitted by the 'rules and controls' (control information). For example, the 'rules and controls' shown in FIG. 2 may grant specific individuals or classes of content users 112 'permission' to use certain content. They may specify what kinds of content usage are permitted, and what kinds are not. They may specify how content usage is to be paid for and how much it costs. As another example, 'rules and controls' may require content usage information to be reported back to the distributor 106 and/or content creator 102." ('193 56:26-35) 12. "ROS VDE functions 604 may be based on segmented, independently loadable executable 'component assemblies' 690. These component assemblies 690 are independently securely deliverable. The component assemblies 690 provided by the preferred embodiment comprise code and data elements that are themselves independently deliverable.... These component assemblies 690 are the basic

	Claim Term/Phrase	Evidence Supporting MS Construction
		functional unit provided by ROS 602. The component assemblies 690 are executed to perform operating system or application tasks. Thus, some component assemblies 690 may be considered to be part of the ROS operating system 602, while other component assemblies may be considered to be 'applications' that run under the support of the operating system." ('193 83:12-29) 13. "As mentioned above, ROS 602 provides several layers of security to ensure the security of component assemblies 690. One important security layer involves ensuring that certain component assemblies 690 are formed, loaded and executed only in secure execution space such as provided within an SPU 500." ('193 87:33-38) 14. "Methods 1000 perform the basic function of defining what users (including, where appropriate, distributions, client administration, etc.), can and cannot do with an object 300." ('193 128:30-33) 15. "Container 152 in this example further includes an electronic control set 188 describing conditions under which the power may be exercised. Controls 188 define the power(s) granted to each of the participants – including (in this example) conditions or limitations for exercising these powers. Controls 188 may provide the same powers and/or conditions of use for each participant, or they may provide different powers and/or conditions of use for each participant." ('712 220:1-8) 16. "...content creators and rights owners can register permissions with the rights and permissions clearinghouses 400 in the form of electronic 'control sets.' These permissions can specify what consumers can and can't do with digital properties, under what conditions the permissions can be exercised and the consequences of exercising the permissions." ('712 72:2-7) 17. "This 'channel 0' 'open channel' task may then issue a series of requests to secure database manager 566 to obtain the 'blueprint' for constructing one or more component assemblies 690 to be associated with channel 594 (block 1127). In the preferred embodiment, this 'blueprint' may comprise a PERC 808 and/or URT 464." ('193 112:46-51) (4) VDE SECURE CONTAINER: See support as listed for Secure Container, item #20, above. **Intrinsic:** 1. "In part, security is enhanced by object methods employed by the present invention because the encryption schemes used to protect an object can efficiently be further used to protect the associated content control information (software control information and relevant data) from modification." ('193 15:51-55) 2. "FIG. 5A shows how the virtual distribution environment 100, in a preferred embodiment, may package information elements (content) into a 'container' 302 so the information can't be accessed except as provided by its 'rules and controls.' Normally, the container 302 is electronic rather than physical. Electronic container 302 in one example comprises 'digital' information having a well defined structure. Container 302 and its contents can be called an 'object

	Claim Term/Phrase	Evidence Supporting MS Construction
		300.'" ('193 58:39-46) 3. "Moreover, when any new VDE object 300 arrives at an electronic appliance 600, the electronic appliance must 'register' the object within object registry 450 so that it can be accessed." ('193 153:56-59) 4. "Even if the object is stored locally to the VDE node, it may be stored as a secure or protected object so that it is not directly accessible to a calling process. ACCESS method 2000 establishes the connections, routings, and security requisites needed to access the object." ('193 192:14-19) 5. "ACCESS method 2000 reads the ACCESS method MDE from the secure database, reads it in accordance with the ACCESS method DTD, and loads encrypted content source and routing information based on the MDE (blocks 2010, 2012). This source and routing information specifies the location of the encrypted content. ACCESS method 2000 then determines whether a connection to the content is available (decision block 2014). This 'connection' could be, for example, an on-line connection to a remote site, a real-time information feed, or a path to a secure/protected resource, for example. If the connection to the content is not currently available ('No' exit of decision block 2014), then ACCESS method 2000 takes steps to open the connection (block 2016). If the connection fails (e.g., because the user is not authorized to access a protected secure resource), then the ACCESS method 2000 returns with a failure indication (termination point 2018)." ('193 192:36-52) 6. "It also employs a software object architecture for VDE content containers that carries protected content and may also carry both freely available information (e.g., summary, table of contents) and secured content control information which ensures the performance of control information." ('193 15:41-46) 7. "In this example, creator 102 may employ one or more application software programs and one or more VDE secure subsystems to place unencrypted content into VDE protected form (i.e., into one or more VDE content containers)." ('193 315:53-56) 8. "The Ginter et al. patent specification referenced above describes many characteristics of this DigiBox™ container model, a powerful, flexible, general construct that enables protected, efficient and interoperable electronic description and regulation of electronic commerce relationships of all kinds..." ('861 1:39-44) 9. "The node and container model described above and in the Ginter et al. patent specification (along with similar other DigiBox/VDE (Virtual Distribution Environment) models) has nearly limitless flexibility." ('861 2:37-40) 10. "Therefore, the container creation and usage tools must themselves be secure in the sense that they must protect certain details about the container design. This additional security requirement can make it even more difficult to make containers easy to use and to provide interoperability." ('861 4:59-64) 11. "FIG. 88 illustrates secure electronic container 302 as an attaché handcuffed to the secure delivery person's wrist. Once again, container is shown as a physical thing for purposes of illustrations only --in the example it is preferably electronic rather than physical, and comprises digital information having a well-defined structure (see FIG. 5A). Special mathematical techniques known as

	Claim Term/Phrase	Evidence Supporting MS Construction
		'cryptography' can be used to make electronic container 302 secure so that only intended recipient 4056 can open the container and access the electronic document (or other items) 4054 it contains." ('683 15:61 - 16:14) 12. "Appliance 600B may deliver the digital copy of item 4054 within a container 302 and/or protect the item with seals. Electronic fingerprints, watermarks and/or other visible and/or hidden markings to provide a 'virtual container' or some of the security or other characteristics of a container (for example, the ability to associate electronic controls with the item). ('683 18:49-56) 13. "For example, defendant's attorney 5052 can specify one container 302 for opening by his co-counsel, client or client in-house counsel, and program another container 302 for opening only by opposing (plaintiff's) counsel 5050. Because of the unique trustedness features provided by system 4050, the defendant's attorney 5052 can have a high degree of trust and confidence that only the authorized parties will be able to open the respective containers and access the information they contain." ('683 56:17-25) 14. "The 'container' concept is a convenient metaphor used to give a name to the collection of elements required to make use of content or to perform an administrative-type activity." ('193 127:30-32) 15. "The virtual distribution environment 100, in a preferred embodiment, may package information elements (content) into a 'container' 302 so the information can't be accessed except as provided by its 'rules and controls.'" ('193 58:39-43) 16. "VDE 100 provides a media independent container model for encapsulating content." ('193 127:2-3) 17. "The electronic form of a document is stored as a VDE container (object) associated with the specific client and/or case. The VDE container mechanism supports a hierarchical ordering scheme for organizing files and other information with a container; this mechanism may be used to organize the electronic copies of the documents within a container, A VDE container is associated with specific access control information and rights that are described in one or more permissions control information sets (PERCs) associated with that container. In this example, only those members of the law firm who possess a VDE instance, an appropriate PERC, and the VDE object that contains the desired document, may use the document." ('193 274:52-64) 18. "The situation is no better for processing documents within the context of ordinary computer and network systems. Although said systems can enforce access control information based on user identity, and can provide auditing mechanism for tracking accesses to files, these are low-level mechanisms that do not permit tracking or controlling the flow of content. In such systems, because document content can be freely copied and manipulated, it is not possible to determine where documents content has gone, or where it came from." ('193 281:27-35) 19. "Secure containers 302 may be used to encapsulate the video and audio being exchanged between electronic kiosk appliances 600, 600' to maintain confidentiality and ensure a high degree of trustedness." ('682 52: 61-64) 20. "[C]ontainer 152 can only be opened within a secure protected processing

	Claim Term/Phrase	Evidence Supporting MS Construction
		environment 154 that is part of the virtual distribution environment described in the above-referenced Ginter et al. patent disclosure ...." ('712 168:22-25) 21. "The present invention provides a new kind of 'virtual distribution environment' (called 'VDE' in this document) that secures, administers, and audits electronic information use. VDE also features fundamentally important capabilities for managing content that travels 'across' the 'information highway.'" ('193 2:24-28) 22. "The present invention truly achieves a content control and auditing architecture that can be configured to most any commercial distribution embodiment." ('193 261:12-15) 23. "The inability of conventional products to be shaped to the needs of electronic information providers and users is sharply in contrast to the present invention. Despite the attention devoted by a cross-section of America's largest telecommunications, computer, entertainment and information provider companies to some of the problems addressed by the present invention, only the present invention provides commercially secure, effective solutions for configurable, general purpose electronic commerce transaction/distribution control systems." ('193 2:13-22) 24. "The configurability provided by the present invention is particularly critical for supporting electronic commerce, that is enabling businesses to create relationships and evolve strategies that offer competitive value. Electronic commerce tools that are not inherently configurable and interoperable will ultimately fail to produce products (and services) that meet both basic requirements and evolving needs of most commerce applications." ('193 16:41-48) 25. "VDE also extends usage control information to an arbitrary granular level (as opposed to a file based level provided by traditional operating systems) ...." ('193 275:8-11) 26. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention: ...." ('193 21:43-45) 27. "A significant facet of the present invention's ability to broadly support electronic commerce is its ability to securely manage independently delivered VDE component objects containing control information ..." ('193 10:66 - 11:2) 28. "Some of the key factors contributing to the configurability intrinsic to the present invention include: ...." ('193 16:66-67) 29. "The scalable transaction management/auditing technology of the present invention will result in more efficient and reliable interoperability ...." ('193 34:9-11) 30. "The present invention answers pressing, unsolved needs by offering a system that supports a standardized control environment which facilitates interoperability of electronic appliances, interoperability of content containers, and efficient creation of electronic commerce applications and models through the use of a programmable, secure electronic transactions management foundation and reusable and extensible executable components." ('193 8:63 - 9:3)

	Claim Term/Phrase	Evidence Supporting MS Construction
		31. "The design of the VDE foundation, VDE load modules, and VDE containers, are important features that enable the VDE node operating environment to be compatible with a very broad range of electronic appliances." ('193 34:26-30) 32. "The ability to optionally incorporate different methods 1000 with each object is important to making VDE 100 highly configurable." ('193 128:28-30) 33. "An important feature of VDE is that it can be used to assure the administration of, and adequacy of security and rights protection for, electronic agreements implemented through the use of the present invention." ('193 46:51-54) 34. "In this example, both the address request 602 and the responsive information 604 are contained within secure electronic containers 152 in order to maintain the confidentiality and integrity of the requests and responses. In this way, for example, outside eavesdroppers cannot tell who sender 95(1) wants to communicate with or what information he or she needs to perform communications with or what information he or she needs to perform the communications – and the directory responses cannot be 'spoofed' to direct the requested message to another location." ('712 12:15-22) 35. "On the other hand, if the information to be exchanged has already been secured and/or is available without authentication (e.g., certain catalog information, containers that have already been encrypted and do not require special handling, etc.), the 'weaker' form of login/password may be used." ('193 290:57-62) 36. "VDE provides means to securely combine content provided at different times, by differing sources, and/or representing different content types. These types, timings, and/or different sources of content can be employed to form a complex array of content within a VDE content container objects, each containing different content whose usage can be controlled, at least in part, by its own container's set of VDE content control information." ('193 297:35-45) 37. "Although methods 1000 can have virtually unlimited variety and some may even be user-defined, certain basic 'use' type methods are preferably used in the preferred embodiment to control most of the more fundamental object manipulation and other functions provided by VDE 100. For example, the following high level methods would typically be provided for object manipulation; OPEN method, READ method, WRITE method, CLOSE method. An OPEN method is used to control opening a container so its content may be accessed. A READ method is used to control access to contents in a container. A WRITE method is used to control the insertion of contents into a container. A CLOSE method is used to close a container that has been opened." ('193 183:12-29) 38. "DESTROY method 2180 removes the ability of a user to use an object by destroying the URT the user requires to access the object. In the preferred embodiment, ... DESTROY method 2180 may than *[sic]* call a WRITE and/or ACCESS method to write information which will corrupt (and thus destroy) the header and/or other important parts of the object (block 2186). DESTROY method 2180 may then mark one or more of the control structures (e.g., the URT) as damaged by writing appropriate information to control structure (blocks 2188, 2190)." ('193 198:41-45) 39. "PANIC method 2200 may prevent the user from further accessing the object

	Claim Term/Phrase	Evidence Supporting MS Construction
		currently being accessed by, for example, destroying the channel being used to access the object and marking one or more of the control structures (e.g., the URT) associated with the user and object as damaged.(blocks 2206, and 2208-2210, respectively). Because the control structure is damaged, the VDE node will need to contact an administrator to obtain a valid control structure(s) before the user may access the same object again." ('193 198:60 - 199:2) 40. "EXTRACT method 2080 is used to copy or remove content from an object and place it into a new object. In the preferred embodiment, the EXTRACT method 2080 does not involve any release of content, but rather simply takes content from one container and places it into another container, both of which may be secure. Extraction of content differs from release in that the content is never exposed outside a secure container." ('193 194:13-20) 41. "Use of secure electronic containers to transport items provides an unprecedented degree of security, trustedness and flexibility." ('683 8:50-52) 42. "Electronic delivery person 4060 can deliver the electronic version of item 4054 within secure container attaché case 302 from personal computer 4116' to another personal computer 4116 operated by recipient 4056." ('683 20:27-30) 43. "Because these transactions are conducted using VDE and VDE secure containers, those observing the communications learn no more than the fact that the parties are communicating." ('712 310:1-3) 44. "VDE in one example provides a 'virtual silicon container' ('virtual black box') in that several different instances of SPU 500 may securely communicate together to provide an overall secure hardware environment that 'virtually' exists at multiple locations and multiple electronic appliances 600. FIG. 87 shows one model 3600 of a virtual silicon container. This virtual container model 3600 includes a content creator 102, a content distributor 106, one or more content redistributors 106a, one or more client administrators 700, one or more client users 3602, and one or more clearinghouses 116. Each of these various VDE participants has an electronic appliance 600 including a protected processing environment 655 that may comprise, at least in part, a silicon-based semiconductor hardware element secure processing unit 500. The various SOUs 500 each encapsulate a part of the virtual distribution environment, and thus, together form the virtual silicon container 3600." ('193 317:58 - 318:8) 45. "Uses tools to transform digital information(such as electronic books, databases, computer software and movies) into protected digital packages called 'objects.' Only those consumers (or other along the chain of possession such as redistributor) who receive permission from a distributor 106 can open these packages. VDE packaged content can be constrained by 'rules and control information.'" ('193 254:18-25) 46. "To open VDE package and make use of its content, and end-user must have permission." ('193 254:45-46) 47. "Place unencrypted content into VDE protected form (i.e., into one or more VDE content containers)." ('193 315:55-56) (5) NON-CIRCUMVENTABLE: **Intrinsic**:

	Claim Term/Phrase	Evidence Supporting MS Construction
		1. "VDE can protect a collection of rights belonging to various parties having in rights in, or to, electronic information. This information may be at one location or dispersed across (and/or moving between) multiple locations. The information may pass through a 'chain' of distributors and a 'chain' of users. Usage information may also be reported through one or more 'chains' of parties. In general, VDE enables parties that (a) have rights in electronic information, and/or (b) act as direct or indirect agents for parties who have rights in electronic information, to ensure that the moving, accessing, modifying, or otherwise using of information can be securely controlled by rules regarding how, when, where, and by whom such activities can be performed." ('193 6:18-31) 2. "All requirements specified by this derived control information must be satisfied before VDE controlled content can be accessed or otherwise used. ('193 11:8-11) 3. "VDE provides important mechanisms for both enforcing commercial agreements and enabling the protection of privacy rights. VDE can securely deliver information from one party to another concerning the use of commercially distributed electronic content. Even if parties are separated by several 'steps' in a chain (pathway) of handling for such content usage information, such information is protected by VDE through encryption and/or other secure processing. Because of that protection, the accuracy of such information is guaranteed by VDE, and the information can be trusted by all parties to whom it is delivered." ('193 14:29-39) 4. "VDE ensures that certain prerequisites necessary for a given transaction to occur are met. This includes the secure execution of any required load modules and the availability of any required, associated data." ('193 20:27-30) 5. "Required methods (methods listed as required for property and/or appliance use) must be available as specified if VDE controlled content (such as intellectual property distributed within a VDE content container) is to be used." ('193 43:37-41) 6. "Since all secure communications are at least in part encrypted and the processing inside the secure subsystem is concealed from outside observation and interference, the present invention ensures that content control information can be enforced." ('193 46:4-8) 7. "This control information can determine, for example: (1) How and/or to whom electronic content can be provided, for example, how an electronic property can be distributed; (2) How one or more objects and/or properties, or portions of an object or property, can be directly used, such as decrypted, displayed, printed, etc; ...." ('193 46:17-24) 8. "'Hardware' 506 also contains long-term and short-term memories to store information securely so it can't be tampered with." ('193 60:1-3) 9. "A feature of VDE provided by the present invention is that certain one or more methods can be specified as required in order for a VDE installation and/or user to be able to use certain and/or all content." ('193 43:47-50) 10. "The virtual distribution environment 100 prevents use of protected information

	Claim Term/Phrase	Evidence Supporting MS Construction
		except as permitted by the 'rules and controls' (control information)." ('193 56:26-28) 11. "As mentioned above, virtual distribution environment 100 'associates' content with corresponding 'rules and controls,' and prevents the content from being used or accessed unless a set of corresponding 'rules and controls' is available. The distributor 106 doesn't need to deliver content to control the content's distribution. The preferred embodiment can securely protect content by protecting corresponding, usage enabling 'rules and controls' against unauthorized distribution and use." ('193 57:18-26) 12. "Since no one can use or access protected content without 'permission' from corresponding 'rules and controls,' the distributor 106 can control use of content that has already been (or will in the future be) delivered." ('193 57:30-33) 13. "SPU 500 is enclosed within and protected by a 'tamper resistant security barrier' 502. Security barrier 502 separates the secure environment 503 from the rest of the world. It prevents information and processes within the secure environment 503 from being observed, interfered with and leaving except under appropriate secure conditions. Barrier 502 also controls external access to secure resources, processes and information within SPU 500." ('193 59:48-55) 14. "Provides non-repudiation of use and may record specific forms of use such as viewing, editing, extracting, copying, redistributing (including to what one or more parties), and/or saving." ('683 6:46-48) 15. "In general, VDE enables parties that (a) have rights in electronic information, and/or (b) act as direct or indirect agents for parties who have rights in electronic information, to ensure that the moving, accessing, modifying, or otherwise using of information can be securely controlled by rules regarding how, when, where, and by whom such activities can be performed." ('193 6:24-30) 16. "To securely control access and other use, including distribution of records, documents, and notes associated with the case" ('193 274:34-36) 17. "Thus wrapped, a VDE object may be distributed to the recipient without fear of unauthorized access and/or other use." ('193 277:16-17) 18. "These appliances typically include a secure subsystem that can enable control of content use such as displaying, encrypting, decrypting, printing, copying, saving, extracting, embedding, distributing, auditing usage, etc." ('193 9:24-27) 19. "VDE provides a secure, distributed electronic transaction management system for controlling the distribution and/or other usage of electronically provided and/or stored information." ('193 9:36-39) 20. "The control set 404 might permit publisher 168 to add his own additional controls that allow consumer 95 to read the work 166 an unlimited number of time but prevent the consumer from copying or redistributing the work." (712 258: 8-11) 21. "The doctor 5000 may then send container 301(1) to a trusted go-between 4700. ... For example, the trusted go-between 4700 in one example has no access to the content of the container 302(1), but does have a record of a seal of

	Claim Term/Phrase	Evidence Supporting MS Construction
		the contents." ('683 53:40-57) 22. "FIG. 116 shows example steps that may be performed by PPE 650 in response to an 'open' or 'view' event. In this example, PPE 650 may - - upon allowing recipient 4056 to actually interact with the item 4054--...PPE 650 may then release the image 4068I and/or the data 4068D to the application running on electronic appliance 600—electronic fingerprinting or watermarking the released content if appropriate (FIG. 116, block 4625C). ('683 42:38-52) 23. "FIG. 5A shows how the virtual distribution environment 100, in a preferred embodiment, may package information elements (content) into a 'container' 302 so the information can't be accessed except as provided by its 'rules and controls.'" ('193 58:39-43) (6) PEER TO PEER: **Intrinsic:** 1. "Each VDE participant in a VDE pathway of content control information may set methods for some or all of the content in a VDE container, so long as such control information does not conflict with senior control information already in place with respect to: (1) certain or all VDE managed content, (2) certain one or more VDE users and/or groupings of users, (3) certain one or more VDE nodes and/or groupings of nodes, and/or (4) certain one or more VDE applications and/or arrangements." ('193 44:6-17) 2. "All participants of VDE 100 have the innate ability to participate in any role." ('193 256:50-51) 3. "Any VDE user 112 may assign the right to process information or perform services on their behalf to the extent allowed by senior control information." ('193 257:17-20) 4. "PERC and URT structures provide a mechanism that may be used to provide precise electronic representation of rights and the controls associated with those rights. VDE thus provides a 'vocabulary' and mechanism by which users and creators may specify their desires." ('193 245:11-15) (7) COMPREHENSIVE RANGE OF FUNCTIONS: **Intrinsic:** 1. "VDE provides comprehensive and configurable transaction management, metering and monitoring technology." ('193 3:34-35) 2. "VDE may be combined with, or integrated into, many separate computers and/or other electronic appliances. These appliances typically include a secure subsystem that can enable control of content use such as displaying, encrypting, decrypting, printing, copying, saving, extracting, embedding, distributing, auditing usage, etc. The secure subsystem in the preferred embodiment comprises one or more 'protected processing environments', one or more secure databases, and secure 'component assemblies' and other items and processes that need to be kept secured. VDE can, for example, securely control electronic currency, payments, and/or credit management (including electronic credit and/or currency receipt, disbursement, encumbering, and/or allocation) using such a 'secure subsystem.'" ('193 9:22-35)

	Claim Term/Phrase	Evidence Supporting MS Construction
		3. "In addition VDE: (a) is very configurable, modifiable, and re-usable; (b) supports a wide range of useful capabilities that may be combined in different ways to accommodate most potential applications; (c) operates on a wide variety of electronic appliances ranging from hand-held inexpensive devices to large mainframe computers; (d) is able to ensure the various rights of a number of different parties, and a number of different rights protection schemes, simultaneously; (e) is able to preserve the rights of parties through a series of transactions that may occur at different times and different locations; (f) is able to flexibly accommodate different ways of securely delivering information and reporting usage; and (g) provides for electronic analogues to 'real' money and credit, including anonymous electronic cash, to pay for products and services and to support personal (including home) banking and other financial activities." ('193 4:57 - 5:10) 4. "[VDE] can provide efficient, reusable, modifiable, and consistent means for secure electronic content: distribution, usage control, usage payment, usage auditing, and usage reporting." ('193 8:26-29) 5. "VDE offers an architecture that avoids reflecting specific distribution biases, administrative and control perspectives, and content types. Instead, VDE provides a broad-spectrum, fundamentally configurable and portable, electronic transaction control, distributing, usage, auditing, reporting, and payment operating environment." ('193 8:53-58) 6. "The present invention allows content providers and users to formulate their transaction environment to accommodate: (1) desired content models, content control models, and content usage information pathways, (2) a complete range of electronic media and distribution means, (3) a broad range of pricing, payment, and auditing strategies, (4) very flexible privacy and/or reporting models, (5) practical and effective security architectures, and (6) other administrative procedures that together with steps (1) through (5) can enable most 'real world' electronic commerce and data security models, including models unique to the electronic world." ('193 10:11-23) 7. "Because of the breadth of issues resolved by the present invention, it can provide the emerging 'electronic highway' with a single transaction/distribution control system that can, for a very broad range of commercial and data security models, ensure against unauthorized use of confidential and/or proprietary information and commercial electronic transactions." ('193 17:22-28) 8. "A feature of the present invention provides for payment means supporting flexible electronic currency and credit mechanisms, including the ability to securely maintain audit trails reflecting information related to use of such currency or credit.' ('193 33:58-63) 9. "The end-to-end nature of VDE applications, in which content 108 flows in one

	Claim Term/Phrase	Evidence Supporting MS Construction
		direction, generating reports and bills 118 in the other, makes it possible to perform 'back-end' consistency checks." ('193 223:17-20) 10. "By way of non-exhaustive summary, these present inventions provide a highly secure and trusted item delivery and agreement execution services providing the following features and functions: Trustedness and security approaching or exceeding that of a personal trusted courier. Instant or nearly instant delivery. Optional delayed delivery ("store and forward"). Broadcasting to multiple parties. Highly cost effective. Trusted validation of item contents and delivery. Value Added Delivery and other features selectable by the sender and/or recipient. Provides electronic transmission trusted auditing and validating. Allows people to communicate quickly, securely, and confidentially. Communications can later be proved through reliable evidence of the communications transaction--providing non-repudiatable, certain, admissible proof that a particular communications transaction occurred. Provides non-repudiation of use and may record specific forms of use such as viewing, editing, extracting, copying, redistributing (including to what one or more parties), and/or saving. Supports persistent rights and rules based document workflow management at recipient sites. System may operate on the Internet, on internal organization and/or corporate networks ("intranets" irrespective of whether they use or offer Internet services internally), private data networks and/or using any other form of electronic communications. System may operate in non-networked and/or intermittently networked environments. Legal contract execution can be performed in real time, with or without face to face or ear-to-ear personal interactions (such as audiovisual teleconferencing, automated electronic negotiations, or any combination of such interactions) for any number of distributed individuals and/or organizations using any mixture of interactions. The items delivered and/or processed may be any 'object' in digital format, including, but not limited to, objects containing or representing data types such as text, images, video, linear motion pictures in digital format, sound recordings and other audio information, computer software, smart agents, multimedia, and/or objects any combination of two or more data types contained within or representing a single compound object. Content (executables for example) delivered with proof of delivery and/or execution or other use. Secure electronic containers can be delivered. The containers can maintain control, audit, receipt and other information and protection securely and persistently in association with one or more items.

	Claim Term/Phrase	Evidence Supporting MS Construction
		Trustedness provides non-repudiation for legal and other transactions. Can handle and send any digital information (for example, analog or digital information representing text, graphics, movies, animation, images, video, digital linear motion pictures, sound and sound recordings, still images, software computer programs or program fragments, executables, data, and including multiple, independent pieces of text; sound clips, software for interpreting and presenting other elements of content, and anything else that is electronically representable). Provides automatic electronic mechanisms that associate transactions automatically with other transactions. System can automatically insert or embed a variety of visible or invisible 'signatures' such as images of handwritten signatures, seals, and electronic 'fingerprints' indicating who has 'touched' (used or other interacted with in any monitorable manner) the item. System can affix visible seals on printed items such as documents for use both in encoding receipt and other receipt and/or usage related information and for establishing a visible presence and impact regarding the authenticity, and ease of checking the authenticity, of the item. Seals can indicate who originated, sent, received, previously received and redistributed, electronically view, and/or printed and/or otherwise used the item. Seals can encode digital signatures and validation information providing time, location, send and/or other information and/or providing means for item authentication and integrity check. Scanning and decoding of item seals can provide authenticity/integrity check of entire item(s) or part of an item (e.g., based on number of words, format, layout, image--picture and/or test--composition, etc.). Seals can be used to automatically associate electronic control sets for use in further item handling. System can hide additional information within the item using 'stenanography' for later retrieval and analysis. Steganography can be used to encode electronic fingerprints and/or other information into an item to prevent deletion. Multiple stenanographic storage of the same fingerprint information may be employed reflecting 'more' public and 'less' public modes so that a less restricted steganographic mode (different encryption algorithm, keys, and/or embedding techniques) can be used to assist easy recognition by an authorized party and a more private (confidential) mode may be readable by only a few parties (or only one party) and comprise of the less restricted mode may not affect the security of the more private mode. Items such as documents can be electronically, optically scanned at the sender's end--and printed out in original, printed form at the recipient's end. Document handlers and processors can integrate document scanning and delivery. Can be directly integrated into enterprise and Internet (and similar network) wide document workflow systems and applications. Secure, tamper-resistant electronic appliance, which may employ VDE SPUs,

	Claim Term/Phrase	Evidence Supporting MS Construction
		used to handle items at both sender and recipient ends. 'Original' item(s) can automatically be destroyed at the sender's end and reconstituted at the recipient's end to prevent two originals from existing simultaneously. Secure, non-repudiable authentication of the identification of a recipient before delivery using any number of different authentication techniques including but not limited to biometric techniques (such as palm print scan, signature scan, voice scan, retina scan, iris scan, biometric fingerprint and/or handprint scan, and/or face profile) and/or presentation of a secure identity 'token.' Non-repudiation provided through secure authentication used to condition events (e.g., a signature is affixed onto a document only if the system securely authenticates the sender and her intention to agree to its contents). Variety of return receipt options including but not limited to a receipt indicating who opened a document, when, where, and the disposition of the document (stored, redistributed, copied, etc.). These receipts can later be used in legal proceedings and/or other contexts to prove item delivery, receipt and/or knowledge. Audit, receipt, and other information can be delivered independently from item delivery, and become securely associated with an item within a protected processing environment. Secure electronic controls can specify how an item is to be processed or otherwise handled (e.g., document can't be modified, can be distributed only to specified persons, collections of persons, organizations, can be edited only by certain persons and/or in certain manners, can only be viewed and will be 'destroyed' after a certain elapse of time or real time or after a certain number of handlings, etc.) Persistent secure electronic controls can continue to supervise item workflow even after it has been received and 'read.' Use of secure electronic containers to transport items provides an unprecedented degree of security, trustedness and flexibility. Secure controls can be used in conjunction with digital electronic certificates certifying as to identity, class (age, organization membership, jurisdiction, etc.) of the sender and/or receiver and/or user of communicated information. Efficiently handles payment and electronic addressing arrangements through use of support and administrative services such as a Distributed Commerce Utility as more fully described in the copending Shear, et al. application. Compatible with use of smart cards, including, for example, VDE enabled smart cards, for secure personal identification and/or for payment. Transactions may be one or more component transactions of any distributed chain of handling and control process including Electronic Data Interchange (EDI) system, electronic trading system, document workflow sequence, and banking and other financial communication sequences, etc." ('683 6:18 - 9:4) 11. "Content providers and distributors have devised a number of limited function rights protection mechanisms to protect their rights. Authorization passwords and protocols, license servers, 'lock/unlock' distribution methods, and non-electronic contractual limitations imposed on users of shrink-wrapped software

	Claim Term/Phrase	Evidence Supporting MS Construction
		are a few of the more prevalent content protection schemes. In a commercial context, these efforts are inefficient and limited solutions." ('193 3:1-9) (8) USER-CONFIGURABLE: **Intrinsic:** 1. "The inability of conventional products to be shaped to the needs of electronic information providers and users is sharply in contrast to the present invention. Despite the attention devoted by a cross-section of America's largest telecommunications, computer, entertainment and information provider companies to some of the problems addressed by the present invention, only the present invention provides commercially secure, effective solutions for configurable, general purpose electronic commerce transaction/distribution control systems." ('193 2:13-22) 2. "The features of VDE allow it to function as the first trusted electronic information control environment that can conform to, and support, the bulk of conventional electronic commerce and data security requirements. In particular, VDE enables the participants in a business value chain model to create an electronic version of traditional business agreement terms and conditions and further enables these participants to shape and evolve their electronic commerce models as they believe appropriate to their business requirements." ('193 8:43-52) 3. "An objective of VDE is supporting a transaction/distribution control standard. Development of such a standard has many obstacles, given the security requirements and related hardware and communications issues, widely differing environments, information types, types of information usage, business and/or data security goals, varieties of participants, and properties of delivered information. A significant feature of VDE accommodates the many, varying distribution and other transaction variables by, in part, decomposing electronic commerce and data security functions into generalized capability modules executable within a secure hardware SPU and/or corresponding software subsystem and further allowing extensive flexibility in assembling, modifying, and/or replacing, such modules (e.g. load modules and/or methods) in applications run on a VDE installation foundation. This configurability and reconfigurability allows electronic commerce and data security participants to reflect their priorities and requirements through a process of iteratively shaping an evolving extended electronic agreement (electronic control model)." ('193 15:66 - 16:18) 4. "Some of the key factors contributing to the configurability intrinsic to the present invention include: (a) integration into the fundamental control environment of a broad range of electronic appliances through portable API and programming language tools that efficiently support merging of control and auditing capabilities in nearly any electronic appliance environment while maintaining overall system security; (b) modular data structures; (c) generic content model;

	Claim Term/Phrase	Evidence Supporting MS Construction
		(d) general modularity and independence of foundation architectural components; (e) modular security structures; (f) variable length and multiple branching chains of control; and (g) independent, modular control structures in the form of executable load modules that can be maintained in one or more libraries, and assembled into control methods and models, and where such model control schemes can 'evolve' as control information passes through the VDE installations of participants of a pathway of VDE content control information handling." ('193 16:66 - 17:21) 5. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention: ... VDE employs a variety of capabilities that serve as a foundation for a general purpose, sufficiently secure distributed electronic commerce solution. VDE enables an electronic commerce marketplace that supports divergent, competitive business partnerships, agreements, and evolving overall business models. For example, ... provide mechanisms that allow control information to 'evolve' and be modified according, at least in part, to independently, securely delivered further control information.... Handlers in a pathway of handling of content control information, to the extent each is authorized, can establish, modify, and/or contribute to, permission, auditing, payment, and reporting control information related to controlling, analyzing, paying for, and/or reporting usage of, electronic content and/or appliances (for example, as related to usage of VDE controlled property content)." ('193 21:43-46; 29:21-41) 6. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention: ... VDE employs a variety of capabilities that serve as a foundation for a general purpose, sufficiently secure distributed electronic commerce solution. VDE enables an electronic commerce marketplace that supports divergent, competitive business partnerships, agreements, and evolving overall business models. For example, ... enable a user to securely extract, through the use of the secure subsystem at the user's VDE installation, at least a portion of the content included within a VDE content container to produce a new, secure object (content container), such that the extracted information is maintained in a continually secure manner through the extraction process." ('193 21:43-46; 31:66 - 32:5) 7. "As with the content control information for most VDE managed content, features of the present invention allows [sic] the content's control information to: (a) 'evolve,' for example, the extractor of content may add new control methods and/or modify control parameter data, such as VDE application compliant methods, to the extent allowed by the content's in-place control information. ... (b) allow a user to combine additional content with at least a portion of said extracted content, ... (c) allow a user to securely edit at least a portion of said content while maintaining said content in a secure form within said VDE content container; ... (d) append extracted content to a pre-existing VDE content container object and attach associated control information ... (e) preserve VDE control over one or more portions of extracted content after

	Claim Term/Phrase	Evidence Supporting MS Construction
		various forms of usage of said portions ... Generally, the extraction features of the present invention allow users to aggregate and/or disseminate and/or otherwise use protected electronic content information extracted from content container sources while maintaining secure VDE capabilities thus preserving the rights of providers in said content information after various content usage processes." ('193 32:27 - 33:4) 8. "The secure component based architecture of ROS 602 has important advantages. For example, it accommodates limited resource execution environments such as provided by a lower cost SPU 500. It also provides an extremely high level of configurability. In fact, ROS 602 will accommodate an almost unlimited diversity of content types, content provider objectives, transaction types and client requirements. In addition, the ability to dynamically assemble independently deliverable components at execution time based on particular objects and users provides a high degree of flexibility" ('193 87:63 - 88:7) 9. "Each logical object structure 800 may also include a 'private body' 806 containing or referencing a set of methods 1000 (i.e., programs or procedures) that control use and distribution of the object 300. The ability to optionally incorporate different methods 1000 with each object is important to making VDE 100 highly configurable." ('193 128:25-30) 10. "VDE methods 1000 are designed to provide a very flexible and highly modular approach to secure processing." ('193 181:17-18) 11. "The reusable functional primitives of VDE 100 can be flexibly combined by content providers to reflect their respective distribution objectives." ('193 255:27-29) 12. "The present invention truly achieves a content control and auditing architecture that can be configured to most any commercial distribution embodiment." ('193 261:12-15) 13. "Adding new content to objects is an important aspect of authoring provided by the present invention. Providers may wish to allow one or more users to add, hide, modify, remove and/or extend content that they provide. In this way, other users may add value to, alter for a new purpose, maintain, and/or otherwise change, existing content. The ability to add content to an empty and/or newly created object is important as well." ('193 261:23-30) 14. "The distribution control information provided by the present invention allows flexible positive control. No provider is required to include any particular control, or use any particular strategy, except as required by senior control information. Rather, the present invention allows a provider to select from generic control components (which may be provided as a subset of components appropriate to a provider's specific market, for example, as included in and/or directly compatible with, a VDE application) to establish a structure appropriate for a given chain of handling/control." ('193 263:9-19)" 15. "Importantly, VDE securely and flexibly supports editing the content in, extracting content from, embedding content into, and otherwise shaping the content composition of, VDE content containers. Such capabilities allow VDE supported product models to evolve by progressively reflecting the

	Claim Term/Phrase	Evidence Supporting MS Construction
		requirements of 'next' participants in an electronic commercial model." ('193 297:9-15) 16. "For instance, the user may have an 'access'right, and an 'extraction' right, but not a 'copy' right." ('193 159:24-26) 17. "PERCS 808 specify a set of rights that may be exercised to use or access the corresponding VDE object 300. The preferred embodiment allows users to 'customize' their access rights by selecting a subset of rights authorized by a corresponding PERC 808 and/or by specifying parameters or choices that correspond to some or all of the rights granted by PERC 808. These user choices are set forth in a user rights table 464 in the preferred embodiment. User rights table (URT) 464 includes URT records, each of which correspond to a user (or group of users). Each of these URT records specific users choices for a corresponding VDE object more methods 1000 for exercising the rights granted to the user by the PERC 808 in a way specified by the choices contained within the URT record." ('193 156:55 - 157:3) 18. "PERC and URT structures provide a mechanism that may be used to provide precise electronic representation of rights and the controls associated with those rights. VDE thus provides a 'vocabulary' and mechanism by which users and creators may specify their desires." ('193 245:10-15) 19. "In sum, the present invention allows information contained in electronic information products to be supplied according to user specification. Tailoring to user specification allows the present invention to provide the greatest value to users, which in turn will generate the greatest amount of electronic commerce activity." ('193 22:66 - 23:5) 20. "Adding new content to objects is an important aspect of authoring provided by the present invention. Providers may wish to allow one or more users to add, hide, modify, remove and/or extend content that they provide. In this way, other users may add value to, alter for a new purpose, maintain, and/otherwise change, existing content. The ability to add content to an empty and/or newly created object is important as well." ('193 261:23-30) 21. "Each logical object structure 800 may also include a 'private body' 806 containing or referencing a set of method 1000 (i.e., programs or procedures) that control use and distribution of the object 300. The ability to optionally incorporate different methods 1000 with each object is important to making VDE 100 highly configurable." ('193 128:25-30) 22. "An important aspect of adding or modifying content is the choice of encryption/decryption keys and/or other relevant aspects of securing new or altered content." ('193 262:21-23) (9) <u>GENERAL PURPOSE; UNIVERSAL</u>: **Intrinsic:** 1. "VDE also features fundamentally important capabilities for managing content that travels 'across' the 'information highway.' These capabilities comprise a rights protection solution that serves all electronic community members. These members include content creators and distributors, financial service providers, end-users, and others. VDE is the first general purpose,

	Claim Term/Phrase	Evidence Supporting MS Construction
		configurable, transaction control/rights protection solution for users of computers, other electronic appliances, networks, and the information highway." ('193 2:27-36) 2. "VDE provides a unified solution that allows all content creators, providers, and users to employ the same electronic rights protection solution." ('193 5:17-19) 3. "Since different groups of components can be put together for different applications, the present invention can provide electronic control information for a wide variety of different products and markets. This means the present invention can provide a 'unified,' efficient, secure, and cost-effective system for electronic commerce and data security. This allows VDE to serve as a single standard for electronic rights protection, data security, and electronic currency and banking." ('193 7:6-14) 4. "Employing VDE as a general purpose electronic transaction/distribution control system allows users to maintain a single transaction management control arrangement on each of their computers, networks, communication nodes, and/or other electronic appliances. Such a general purpose system can serve the needs of many electronic transaction management applications without requiring distinct, different installations for different purposes. As a result, users of VDE can avoid the confusion and expense and other inefficiencies of different, limited purpose transaction control applications for each different content and/or business model. For example, VDE allows content creators to use the same VDE foundation control arrangement for both content authoring and for licensing content from other content creators for inclusion into their products or for other use. Clearinghouses, distributors, content creators, and other VDE users can all interact, both with the applications running on their VDE installations, and with each other, in an entirely consistent manner, using and reusing (largely transparently) the same distributed tools, mechanisms, and consistent user interfaces, regardless of the type of VDE activity." ('193 11:38-59) 5. "An objective of VDE is supporting a transaction/distribution control standard." ('193 15:66-67) 6. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... The design of the VDE foundation, VDE load modules, and VDE containers, are important features that enable the VDE node operating environment to be compatible with a very broad range of electronic appliances. The ability, for example, for control methods based on load modules to execute in very 'small' and inexpensive secure sub-system environments, such as environments with very little read/write memory, while also being able to execute in large memory sub-systems that may be used in more expensive electronic appliances, supports consistency across many machines. This consistent VDE operating environment, including its control structures and container architecture, enables the use of standardized VDE content containers across a broad range of device types and host operating environments. Since VDE capabilities can be seamlessly integrated as extensions, additions, and/or modifications to fundamental capabilities of

	Claim Term/Phrase	Evidence Supporting MS Construction
		electronic appliances and host operating systems, VDE containers, content control information, and the VDE foundation will be able to work with many device types and these device types will be able to consistently and efficiently interpret and enforce VDE control information." ('193 21:43-46; 34:26-49) 7. "This rationalization stems from the reusability of control structures and user interfaces for a wide variety of transaction management related activities. As a result, content usage control, data security, information auditing, and electronic financial activities, can be supported with tools that are reusable, convenient, consistent, and familiar. In addition, a rational approach--a transaction/distribution control standard--allows all participants in VDE the same foundation set of hardware control and security, authoring, administration, and management tools to support widely varying types of information, business market model, and/or personal objectives." ('193 11:26-37) 8. "Because of the breadth of issues resolved by the present invention, it can provide the emerging 'electronic highway' with a single transaction/ distribution control system that can, for a very broad range of commercial and data security models, ensure against unauthorized use of confidential and/or proprietary information and commercial electronic transactions. VDE's electronic transaction management mechanisms can enforce the electronic rights and agreements of all parties participating in widely varying business and data security models, and this can be efficiently achieved through a single VDE implementation within each VDE participant's electronic appliance. VDE supports widely varying business and/or data security models that can involve a broad range of participants at various 'levels' of VDE content and/or content control information pathways of handling. Different content control and/or auditing models and agreements may be available on the same VDE installation. These models and agreements may control content in relationship to, for example, VDE installations and/or users in general; certain specific users, installations, classes and/or other groupings of installations and/or users; as well as to electronic content generally on a given installation, to specific properties, property portions, classes and/or other groupings of content." ('193 17:22-45) 9. "The present invention's trusted/secure, universe wide, distributed transaction control and administration system." ('193 35:66 - 36:1) 10. "Commerce Utility Systems 90 are generalized and programmable..." ('712 67:7-8) (10) FLEXIBLE: **Intrinsic:** 1. "Providers of 'electronic currency' have also created protections for their type of content. These systems are not sufficiently adaptable, efficient, nor flexible enough to support the generalized use of electronic currency. Furthermore, they do not provide sophisticated auditing and control configuration capabilities. This means that current electronic currency tools lack the sophistication needed

	Claim Term/Phrase	Evidence Supporting MS Construction
		for many real-world financial business models. VDE provides means for anonymous currency and for 'conditionally' anonymous currency, wherein currency related activities remain anonymous except under special circumstances." ('193 3:10-20) 2. "Traditional content control mechanisms often require users to purchase more electronic information than the user needs or desires. For example, infrequent users of shrink-wrapped software are required to purchase a program at the same price as frequent users, even though they may receive much less value from their less frequent use. Traditional systems do not scale cost according to the extent or character of usage and traditional systems can not attract potential customers who find that a fixed price is too high. Systems using traditional mechanisms are also not normally particularly secure. For example, shrink-wrapping does not prevent the constant illegal pirating of software once removed from either its physical or electronic package." ('193 5:50-62) 3. "Traditional electronic information rights protection systems are often inflexible and inefficient and may cause a content provider to choose costly distribution channels that increase a product's price. In general these mechanisms restrict product pricing, configuration, and marketing flexibility. These compromises are the result of techniques for controlling information which cannot accommodate both different content models and content models which reflect the many, varied requirements, such as content delivery strategies, of the model participants. This can limit a provider's ability to deliver sufficient overall value to justify a given product's cost in the eyes of many potential users. VDE allows content providers and distributors to create applications and distribution networks that reflect content providers' and users' preferred business models. It offers users a uniquely cost effective and feature rich system that supports the ways providers want to distribute information and the ways users want to use such information." ('193 5:63 - 6:13) 4. "VDE does not require electronic content providers and users to modify their business practices and personal preferences to conform to a metering and control application program that supports limited, largely fixed functionality. Furthermore, VDE permits participants to develop business models not feasible with non- electronic commerce, for example, involving detailed reporting of content usage information, large numbers of distinct transactions at hitherto infeasible low price points, 'pass-along' control information that is enforced without involvement or advance knowledge of the participants, etc." ('193 9:67 - 10:9) 5. "VDE can further be used to enable commercially provided electronic content to be made available to users in user defined portions, rather than constraining the user to use portions of content that were 'predetermined' by a content creator and/or other provider for billing purposes." ('193 11:66 - 12:4) 6. "The 'usage map' concept provided by the preferred embodiment may be tied to the concept of 'atomic elements.' In the preferred embodiment, usage of an object 300 may be metered in terms of 'atomic elements.' In the preferred embodiment, an 'atomic element' in the metering context defines a unit of usage that is 'sufficiently significant' to be recorded in a meter. The definition

	Claim Term/Phrase	Evidence Supporting MS Construction
		of what constitutes an 'atomic element' is determined by the creator of an object 300. For instance, a 'byte' of information content contained in an object 300 could be defined as an 'atomic element,' or a record of a database could be defined as an 'atomic element,' or each chapter of an electronically published book could be defined as an 'atomic element.'" ('193 144:53-65) 7. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention. VDE employs a variety of capabilities that serve as a foundation for a general purpose, sufficiently secure distributed electronic commerce solution. VDE enables an electronic commerce marketplace that supports divergent, competitive business partnerships, agreements, and evolving overall business models. For example, VDE includes features that. . . support dynamic user selection of information subsets of a VDE electronic information product (VDE controlled content). This contrasts with the constraints of having to use a few high level individual, pre-defined content provider information increments such as being required to select a whole information product or product section in order to acquire or otherwise use a portion of such product or section. VDE supports metering and usage control over a variety of increments (including 'atomic' increments, and combinations of different increment types) that are selected ad hoc by a user and represent a collection of pre-identified one or more increments (such as one or more blocks of a preidentified nature, e.g., bytes, images, logically related blocks) that form a generally arbitrary, but logical to a user, content 'deliverable.' VDE control information (including budgeting, pricing and metering) can be configured so that it can specifically apply, as appropriate, to ad hoc selection of different, unanticipated variable user selected aggregations of information increments and pricing levels can be, at least in part, based on quantities and/or nature of mixed increment selections (for example, a certain quantity of certain text could mean associated images might be discounted by 15%; a greater quantity of text in the 'mixed' increment selection might mean the images are discounted 20%). Such user selected aggregated information increments can reflect the actual requirements of a user for information and is more flexible than being limited to a single, or a few, high level, (e.g. product, document, database record) predetermined increments. Such high level increments may include quantities of information not desired by the user and as a result be more costly than the subset of information needed by the user if such a subset was available. In sum, the present invention allows information contained in electronic information products to be supplied according to user specification. Tailoring to user specification allows the present invention to provide the greatest value to users, which in turn will generate the greatest amount of electronic commerce activity. The user, for example, would be able to define an aggregation of content derived from various portions of an available content product, but which, as a deliverable for use by the user, is an entirely unique aggregated increment. The user may, for example, select certain numbers of bytes of information from various portions of an information product, such as a reference work, and copy them to disc in unencrypted form and be billed based on total number of bytes plus a surcharge on the number of 'articles' that provided the bytes. A content provider might

	Claim Term/Phrase	Evidence Supporting MS Construction
		reasonably charge less for such a user defined information increment since the user does not require all of the content from all of the articles that contained desired information." ('193 21:43-53; 22:32-49) 8. "Summary of Some Important Features Provided by VDE in Accordance With the Present Invention ... Differing models for billing, auditing, and security can be applied to the same piece of electronic information content and such differing sets of control information may employ, for control purposes, the same, or differing, granularities of electronic information control increments." ('193 21:43-46; 28:23-28) 9. "The VDE templates, classes, and control structures are inherently flexible and configurable to reflect the breadth of information distribution and secure storage requirements, to allow for efficient adaptation into new industries as they evolve, and to reflect the evolution and/or change of an existing industry and/or business, as well as to support one or more groups of users who may be associated with certain permissions and/or budgets and object types. The flexibility of VDE templates, classes, and basic control structures is enhanced through the use of VDE aggregate and control methods which have a compound, conditional process impact on object control. Taken together, and employed at times with VDE administrative objects and VDE security arrangements and processes, the present invention truly achieves a content control and auditing architecture that can be configured to most any commercial distribution embodiment. Thus, the present invention fully supports the requirements and biases of content providers without forcing them to fit a predefined application model. It allows them to define the rights, control information, and flow of their content (and the return of audit information) through distribution channels." ('193 260:66 - 261:20) 10. "VDE also extends usage control information to an arbitrary granular level (as opposed to a file based level provided by traditional operating systems) and provides flexible control information over any action associated with the information which can be described as a VDE controlled process." ('193 275:8-13) 11. "The situation is no better for processing documents within the context of ordinary computer and network systems. Although said systems can enforce access control information based on user identity, and can provide auditing mechanisms for tracking accesses to files, these are low-level mechanisms that do not permit tracking or controlling the flow of content. In such systems, because document content can be freely copied and manipulated, it is not possible to determine where document content has gone, or where it came from. In addition, because the control mechanisms in ordinary computer operating systems operate at a low level of abstraction, the entities they control are not necessarily the same as those that are manipulated by users. This particularly causes audit trails to be cluttered with voluminous information describing uninteresting activities." ('193 281:27-41) 12. "Importantly, VDE securely and flexibly supports editing the content in, extracting content from, embedding content into, and otherwise shaping the content composition of, VDE content containers." ('193 297:9-12)

	Claim Term/Phrase	Evidence Supporting MS Construction
		13. "The InterTrust DigiBox container model allows and facilitates these and other different container uses. It facilitates detailed container customization for different uses, classes of use and/or users in order to meet different needs and business models. This customization ability is very important, particularly when used in conjunction with a general purpose, distributed rights management environment such as described in Ginter, et al. Such an environment calls for a practical optimization of customizability, including customizability and transparency for container models. This customization flexibility has a number of advantages, such as allowing optimization (e.g., maximum efficiency, minimum overhead) of the detailed container design for each particular application or circumstance so as to allow many different container designs for many different purposes (e.g., business models) to exist at the same time and be used by the rights control client (node) on a user electronic appliance such as a computer or entertainment device." ('861 2:49-67) 14. "The node and container model described above and in the Ginter et al. patent specification (along with similar other DigiBox/VDE (Virtual Distribution Environment) models) has nearly limitless flexibility." ('861 2:37-40) 15. "Such capabilities allow VDE supported product models to evolve by progressively reflecting requirements of 'next' participants in an electronic commercial models." ('193 297:12-15)
25.	193.1: "a budget specifying the number of copies which can be made of said digital file"	**Intrinsic:** 1. "For example, content control information for a given piece of content may be stipulated as senior information and therefore not changeable, might be put in place by a content creator and might stipulate that national distributors of a given piece of their content may be permitted to make 100,000 copies per calendar quarter, so long as such copies are provided to bonfire end-users, but may pass only a single copy of such content to a local retailers and the control information limits such a retailer to making no more than 1,000 copies per month for retail sales to end-users. In addition, for example, an end-user of such content might be limited by the same content control information to making three copies of such content, one for each of three different computers he or she uses (one desktop computer at work, one for a desktop computer at home, and one for a portable computer)." ('193 48:19-34) 2. "... storing a first digital file and a first control in a first secure container, said first control constituting a first budget which governs the number of copies which may be made of said first digital file or a portion of said first digital file while said first digital file is contained in said first secure container," ('193 claim 60) 3. "A certain content provider might, for example, require metering the number of copies made for distribution to employees of a given software program (a portion of the program might be maintained in encrypted form and require the presence of a VDE installation to run). This would require the execution of a metering method for copying of the property each time a copy was made for another employee." ('193 20:36-43) 4. "For example, in the earlier example of a user with a desktop and a notebook computer, a provider may allow a user to make copies of information necessary to enable the notebook computer based on information present in the desktop

	Claim Term/Phrase	Evidence Supporting MS Construction
		computer, but not allow any further copies of said information to be made by the notebook VDE node. In this example, the distribution control structure described earlier would continue to exist on the desktop computer, but the copies of the enabling information passed to the notebook computer would lack the required distribution control structure to perform distribution from the notebook computer. Similarly, a distribution control structure may be provided by a content provider to a content provider who is a distributor in which a control structure would enable a certain number of copies to be made of a VDE content container object along with associated copies of permissions records, but the permissions records would be altered (as per specification of the content provider, for example) so as not to allow end-users who received distributor created copies from making further copies for distribution to other VDE nodes." ('193 264:29-49) 5. "SPU 500 is enclosed within and protected by a 'tamper resistant security barrier' 502. Security barrier 502 separates the secure environment 503 from the rest of the world. It prevents information and processes within the secure environment 503 from being observed, interfered with and leaving except under appropriate secure conditions." ('193 59:48-53) 6. "Secure container 302 may also contain an electronic, digital control structure 4078. This control structure 4078 (which could also be delivered independently in another container 302 different from the one carrying the image 4068I and/or the data 4068D) may contain important information controlling use of container 302. For example, controls 4078 may specify who can open container 302 and under what conditions the container can be opened. Controls 4078 might also specify who, if anyone, object 300 can be passed on to. As another example, controls 4078 might specify restrictions on how the image 4068I and/or data 4068D can be used (e.g., to allow the recipient to view but not change the image and/or data as one example). The detailed nature of control structure 4078 is described in connection, for example, with FIGS. 11D-11J ; FIG. 15 ; FIGS. 17-26B; and FIGS. 41A-61." ('683 25:62–26:10) 7. "Many objects 300 that are distributed by physical media and/or by 'out of channel' means (e.g., redistributed after receipt by a customer to another customer) might not include key blocks 810 in the same object 300 that is used to transport the content protected by the key blocks. This is because VDE objects may contain data that can be electronically copied outside the confines of a VDE node. If the content is encrypted, the copies will also be encrypted and the copier cannot gain access to the content unless she has the appropriate decryption key(s)." ('193 128:66) 8. "Although block 1262 includes encrypted summary services information on the back up, it preferably does not include SPU device private keys, shared keys, SPU code and other internal security information to prevent this information from ever becoming available to users even in encrypted form." ('193 166:59-64)
26.	193.1: "controlling the copies made of said digital file"	See above.

	Claim Term/Phrase	Evidence Supporting MS Construction
27.	721.1: "digitally signing a second load module with a second digital signature different from the first digital signature, the second digital signature designating the second load module for use by a second device class having at least one of tamper resistance and security level different from the at least one of tamper resistance and security level of the first device class"	**Intrinsic:** 1. "In one example, verifying authority 100 may digitally sign identical copies of load module 54 for use by different classes or 'assurance levels' of electronic appliances 61." ('721 18:19-22) 2. "Protected execution spaces such as protected processing environments can be programmed or otherwise conditioned to accept only those load modules or other executables bearing a digital signature/certificate of an accredited (or particular) verifying authority. Tamper resistant barriers may be used to protect this programming or other conditioning. The assurance levels described below are a measure or assessment of the effectiveness with which this programming or other conditioning is protected." ('721 5:1-9) 3. "For example, protected processing environments or other secure execution spaces that are more impervious to tampering (such as those providing a higher degree of physical security) may use an assurance level that isolates it from protected processing environments or other secure execution spaces that are relatively more susceptible to tampering (such as those constructed solely by software executing on a general purpose digital computer in a non-secure location)." ('721 6:34-41) 4. "The present invention may use a verifying authority and the digital signatures it provides to compartmentalize the different electronic appliances depending on their level of security (e.g., work factor or relative tamper resistance)." ('721 6:53-56) 5. "Assurance level I might be used for an electronic appliance(s) 61 whose protected processing environment 108 is based on software techniques that may be somewhat resistant to tampering. An example of an assurance level I electronic appliance 61A might be a general purpose personal computer that executes software to create protected processing environment 108. An assurance level II electronic appliance 61B may provide a protected processing environment 108 based on a hybrid of software security techniques and hardware-based security techniques. An example of an assurance level II electronic appliance 61B might be a general purpose personal computer equipped with a hardware integrated circuit secure processing unit ('SPU') that performs some secure processing outside of the SPU (see Ginter et al. patent disclosure FIG. 10 and associated text). Such a hybrid arrangement might be relatively more resistant to tampering than a software-only implementation. The assurance level III appliance 61C shown is a general purpose personal computer equipped with a hardware-based secure processing unit 132 providing and completely containing protected processing environment 108 (see Ginter et al. FIGS. 6 and 9 for example). A silicon-based special purpose integrated circuit security chip is relatively more tamper-resistant than implementations relying on software techniques for some or all of their tamper-resistance." ('721 6:44 – 7:5) 6. "Assurance level in this example may be assigned to a particular protected processing environment 108 at initialization (e.g., at the factory in the case of hardware-based secure processing units). Assigning assurance level at initialization time facilitates the use of key management (e.g., secure key exchange protocols) to enforce isolation based on assurance level. For example,

	Claim Term/Phrase	Evidence Supporting MS Construction
		since establishment of assurance level is done at initialization time, rather than in the field in this example, the key exchange mechanism can be used to provide new keys (assuming an assurance level has been established correctly)." ('721 17:13-23)
28.	891.1: "securely applying, at said first appliance through use of said at least one resource said first entity's control and said second entity's control to govern use of said data item"	**Intrinsic:** 1. "Such secure combination of VDE manage pieces of content will frequently require VDE's ability to securely derive content control information which accommodates the control information requirements, including any combinational rules, of the respective VDE managed pieces of content and reflects an acceptable agreement between plural control information sets." ('193 296:26-32)
29.	900.155: "derives information from one or more aspects of said host processing environment"	**Intrinsic:** 1. See '900 73:1- 80:6 a. "SPU Integrated Within CPU b. As discussed above, it may be desirable to integrate CPU 654 and SPU 500 into the same integrated circuit and/or device. SPU 500 shown in FIG. 9 includes a microprocessor 520 that may be similar or identical to a standard microprocessor available off-the-shelf from a variety of manufacturers. Similarly, the SPU DMA controller 526 and certain other microprocessor support circuitry may be standard implementations available in off-the-shelf microprocessor and/or microcomputer chips. Since many of the general control and processing requirements provided by SPU 500 in the preferred embodiment can be satisfied using certain generic CPU and/or microcontroller components, it may be desirable to integrate SPU VDE functionality into a standard generic CPU or microcontroller chip. Such an integrated solution can result in a very cost-effective 'dual mode' component that is capable of performing all of the generic processing of a standard CPU as well as the secure processing of an SPU. Many of the control logic functions performed by the preferred embodiment SPU can be performed by generic CPU and/or micro-controller logic so that at least a portion of the control logic does not have to be duplicated. Additional cost savings (e.g., in terms of reducing manufacturing costs, inventory costs and printed circuit board real estate requirements) may also be obtained by not requiring an additional, separate physical SPU 500 device or package. FIG. 9A shows one example architecture of a combination CPU/SPU 2650. CPU/SPU

	Claim Term/Phrase	Evidence Supporting MS Construction
		2650 may include a standard microprocessor or microcontroller 2652, a standard bus interface unit (BIU) 2656, and a standard (optional) DMA controller 2654, as well as various other standard I/O controllers, computation circuitry, etc. as may be found in a typical off-the-shelf microprocessor/microcontroller. Real time clock 528 may be added to the standard architecture to give the CPU/SPU 2650 access to the real time clock functions as discussed above in connection with FIG. 9. Real-time clock 528 must be protected from tampering in order to be secure. Such protections may include internal or external backup power, an indication that its power (and thus its operation) has been interrupted, and/or an indication that the external clock signal(s) from which it derives its timing have been interfered with (e.g., sped up, slowed down). Similarly, an encrypt/decrypt engine 522, pattern matching engine 524, compression/decompression engine 546 and/or arithmetic accelerator 544 may be added if desired to provide greater efficiencies, or the functions performed by these components could be provided instead by software executing on microprocessor 2652. An optional memory management unit 540 may also be provided if desired. A true random number generator 542 may be provided also if desired. Connections shown between mode interface switch 2658 and other components can carry both data and control information, specifically control information that determines what security-relevant aspects of the other components are available for access and/or manipulation. c. In addition, secure ROM 532 and/or secure RAM 534 may be provided within CPU/SPU 2650 along with a 'mode interface switch' 2658a, 2658b. Mode interface switch 2658 selectively provides microprocessor 2652 with access to secure memory 532, 534 and other secure components (blocks 522, 546, 524, 542, 544, 528) depending upon the 'mode' CPU/SPU 2650 is operating in. CPU/SPU 2650 in this example may operate in two different modes: an 'SPU' mode, or a 'normal' mode. In the 'normal' mode, CPU/SPU 2650 operates substantially identically to a standard off-the-shelf CPU while also protecting the security of the content, state, and operations of security-relevant components included in CPU/SPU 2650. Such security-relevant components may include the secure memories 532, 534; the encrypt/decrypt engine 522, the optional pattern-matching engine 524, random number generator 542, arithmetic accelerator 544, the SPU-not-initialized flag 2671, the secure mode interface switch 2658, the real-time clock 528, the DMA controller 2654, the MMU 540, compress/decompress block 546, and/or any other components that may affect security of the operation of the CPU/SPU in 'SPU' mode. d. In this example, CPU/SPU 2650 operating in the 'normal' mode controls mode interface switch 2658 to effectively 'disconnect' (i.e., block unsecure access to) the security-relevant components, or to the security-relevant aspects of the operations of such components as have a function for both 'normal' and 'SPU' mode. In the 'normal' mode, for

	Claim Term/Phrase	Evidence Supporting MS Construction
		example, microprocessor 2652 could access information from standard registers or other internal RAM and/or ROM (not shown), execute instructions in a 'normal' way, and perform any other tasks as are provided within a standard CPU--but could not access or compromise the contents of secure memory 532, 534 or access blocks 522, 524, 542, 544, 546. In this example 'normal' mode, mode interface switch 2658 would effectively prevent any access (e.g., both read and write access) to secure memory 532, 534 so as to prevent the information stored within that secure memory from being compromised. e. When CPU/SPU 2650 operates in the 'SPU' mode, mode interface switch 2658 allows microprocessor 2652 to access secure memory 532, 534, and to control security-relevant aspects of other components in the CPU/SPU. The 'SPU' mode in this example requires all instructions executed by microprocessor 2652 to be fetched from secure memory 532, 534--preventing execution based on 'mixed' secure and non-secure instructions. In the 'SPU' mode, mode interface switch 2658 may, in one example embodiment, disconnect or otherwise block external accesses carried over bus 652 from outside CPU/SPU 2650 (e.g., DMA accesses, cache coherency control accesses) to ensure that the microprocessor 2652 is controlled entirely by instructions carried within or derived from the secure memory 532, 534. Mode interface switch 2658 may also disconnect or otherwise block access by microprocessor 2652 to some external memory and/or other functions carried over bus 652. Mode interface switch 2658 in this example prevents other CPU operations/instructions from exposing the contents of secure memory 532, 534. f. In the example shown in FIG. 9A, the mode control of mode interface switch 2658 is based on a 'mode' control signal provided by microprocessor 2652. In this example, microprocessor 2652 may be slightly modified so it can execute two 'new' instructions: 'enable `SPU` mode' instruction, and 'disable `SPU` mode' instruction. g. When microprocessor 2652 executes the 'enable `SPU` mode' instruction, it sends an appropriate 'mode' control signal to mode interface switch 2658 to 'switch' the interface switch into the 'SPU' mode of operation. When microprocessor 2652 executes the 'disable `SPU` mode' instruction, it sends an appropriate 'mode' control signal to mode interface switch 2658 to disable the 'SPU' mode of operation. h. When CPU/SPU 2650 begins operating in the 'SPU' mode (based on microprocessor 2652 executing the 'enable 'SPU' mode' instruction), mode interface switch 2658 forces microprocessor 2652 to begin fetching instructions from secure memory 532, 534 (e.g., beginning at some fixed address) in one example. When CPU/SPU 2650 begins operating in this example 'SPU' mode, mode interface switch 2658 may force microprocessor 2652 to load its registers from some fixed address in secure memory 532, 534 and may begin execution based on such register content. Once operating in the 'SPU' mode, microprocessor 2652 may

	Claim Term/Phrase	Evidence Supporting MS Construction
		provide encryption/decryption and other control capabilities based upon the code and other content of secure memory 532, 534 needed to provide the VDE functionality of SPU 500 described above. For example, microprocessor 2652 operating under control of information within secure memory 532, 534 may read encrypted information from bus 652 via bus interface unit 2656, write decrypted information to the bus interface unit, and meter and limit decryption of such information based on values stored in the secure memory. i. At the end of secure processing, execution by microprocessor 2652 of the 'disable SPU mode' instruction may cause the contents of all registers and other temporary storage locations used by microprocessor 2652 that are not within secure memory 532, 534 to be destroyed or copied into secure memory 532, 534 before 'opening' mode interface switch 2658. Once mode interface switch 2658 is 'open,' the microprocessor 2652 no longer has access to secure memory 532, 534 or the information it contained, or to control or modify the state of any other security-relevant components or functions contained within CPU/SPU 2650 to which access is controlled by mode interface switch 2658. j. Whenever CPU/SPU 2650 enters or leaves the 'SPU' mode, the transition is performed in such a way that no information contained in the secure memory 532, 534 or derived from it (e.g., stored in registers or a cache memory associated with microprocessor 2652) while in the 'SPU' mode can be exposed by microprocessor 2652 operations that occur in the 'normal' mode. This may be accomplished either by hardware mechanisms that protect against such exposure, software instructions executed in 'SPU' mode that clear, reinitialize, and otherwise reset during such transitions, or a combination of both. k. In some example implementations, interrupts may be enabled while CPU/SPU 2650 is operating in the 'SPU' mode similarly interrupts and returns from interrupts while in the 'SPU' mode may allow transitions from 'SPU' mode to 'normal' mode and back to 'SPU' mode without exposing the content of secure memory 532, 534 or the content of registers or other memory associated with microprocessor 2652 that may contain information derived from secure mode operation. l. In some example implementations, there may be CPU/SPU activities such as DMA transfers between external memory and/or devices and secure memory 532, 534 that are initiated by microprocessor 2652 but involve autonomous activity by DMA controller 2654 and, optionally, encrypt/decrypt engine 522 and/or compress/decompress engine 546. In such implementations, mode interface switch 2658 and its associated control signals may be configured to permit such pending activities (e.g. DMA transfers) to continue to completion even after CPU/SPU 2650 leaves 'SPU' mode, provided that upon completion, all required clearing, reinitialization, and/or reset activities occur, and provided that no access or interference is permitted with the pending activities except when CPU/SPU 2650 is operating in 'SPU' mode.

	Claim Term/Phrase	Evidence Supporting MS Construction
		m. In an additional example embodiment, encryption/decryption logic may be connected between microprocessor 2652 and secure memory 532, 354. This additional encryption/decryption logic may be connected 'in parallel' to mode interface switch 2658. The additional encryption/decryption logic may allow certain accesses by microprocessor 2652 to the secure memory 532, 534 when CPU/SPU 2650 is operating in the 'normal' mode. In this alternate embodiment, reads from secure memory 532, 534 when CPU/SPU 2650 is operating in the 'normal' mode automatically result in the read information being encrypted before it is delivered to microprocessor 2652 (and similarly, and writes to the secure memory may result in the written information being decrypted before it is deposited into the secure memory). This alternative embodiment may permit access to secure memory 532, 534 (which may in this example store the information in 'clear' form) by microprocessor 2652 when CPU/SPU 2650 is operating in the 'non-secure normal' mode, but only reveals the secure memory contents to microprocessor 2652 in unencrypted form when the CPU/SPU is operating in the 'SPU' mode. Such access may also be protected by cryptographic authentication techniques (e.g., message authentication codes) to prevent modification or replay attacks that modify encrypted data stored in secure memory 532, 534. Such protection may be performed utilizing either or both of software and/or hardware cryptographic techniques. n. All of the components shown in FIG. 9A may be disposed within a single integrated circuit package. Alternatively, mode interface switch 2658 and secure memory 532, 534, and other security-relevant components might be placed within an integrated circuit chip package and/or other package separate from the rest of CPU/SPU 2650. In this two-package version, a private bus could be used to connect microprocessor 2652 to the mode interface switch 2658 and associated secure memory 532, 534. To maintain security in such multi-package versions, it may be necessary to enclose all the packages and their interconnections in an external physical tamper-resistant barrier. o. Initialization of Integrated CPU/SPU p. Instructions and/or data may need to be loaded into CPU/SPU 2650 before it can operate effectively as an SPU 500. This may occur during the manufacture of CPU/SPU 2650 or subsequently at a CPU/SPU initialization facility. Security of such initialization may depend on physical control of access to the CPU/SPU component(s), on cryptographic means, or on some combination of both. Secure initialization may be performed in plural steps under the control of different parties, such that an initialization step to be performed by party B is preconditioned on successful performance of a step by party A. Different initialization steps may be protected using different security techniques (e.g. physical access, cryptography). q. In this example, switch 2658 may expose an external control signal 2670 that requests operation in 'SPU' mode rather than 'normal'

	Claim Term/Phrase	Evidence Supporting MS Construction
		mode after a power-on reset. This signal would be combined (e.g., by a logical AND 2672) with a non-volatile storage element 2671 internal to CPU/SPU 2650. If both of these signals are asserted, AND gate 2672 would cause CPU/SPU 2650 to begin operating in SPU mode, either executing existing instructions from an address in SPU memory 532, executing instructions from main memory 2665 or otherwise external to the CPU/SPU. The instructions thus executed would permit arbitrary initialization and other functions to be performed in 'SPU' mode without necessarily requiring any instructions to be previously resident in the SPU memory 532. r. Once initialized, the SPU would, under control of its initialization program, indicate to switch 2658 that the flag 2671 is to be cleared. Clearing flag 2671 would permanently disable this initialization capability because no mechanism would be provided to set flag 2671 back to its initial value. If flag 2671 is clear, or control signal 2670 is not asserted, CPU/SPU 2650 would behave precisely as does microprocessor 2652 with respect to power-on reset and other external conditions. Under such conditions, only execution of the 'enable SPU mode' instruction or otherwise requesting SPU mode under program control would cause 'SPU' mode to be entered. s. Additionally, a mechanism could be provided to permit microprocessor 2652 and/or control signal 2672 to reinitialize the flag 2671. Such reinitialization would be performed in a manner that cleared secure memory 532, 534 of any security-relevant information and reinitialized the state of all security-relevant components. This reinitialization mechanism would permit CPU/SPU 2650 to be initialized several times, facilitating testing and/or re-use for different applications, while protecting all security-relevant aspects of its operation. t. In the preferred embodiment, CPU/SPU 2650 would, when SPU mode has not yet been established, begin operating in SPU mode by fetching instructions from secure non-volatile memory 532, thereby ensuring a consistent initialization sequence and preventing SPU dependence on any information held outside CPU/SPU 2650. This approach permits secret initialization information (e.g., keys for validating digital signatures on additional information to be loaded into secure memory 532, 534) to be held internally to CPU/SPU 2650 so that it is never exposed to outside access. Such information could even be supplied by a hardware 'mask' used in the semiconductor fabrication process. u. CPU/SPU Integrated With Unmodified Microprocessor v. FIG. 9B shows an additional example embodiment, in which a completely standard microprocessor 2652 integrated circuit chip could be transformed into a CPU/SPU 2650 by adding an SPU chip 2660 that mediates access to external I/O devices and memory. In such an embodiment, the microprocessor 2652 would be connected to the SPU chip 2660 by a private memory bus 2661, and all three such components would be contained within hardware tamper-resistant barrier 502.

	Claim Term/Phrase	Evidence Supporting MS Construction
		w. In this embodiment, SPU chip 2660 may have the same secure components as in FIG. 9, i.e., it may have a ROM/EEPROM 532, a RAM 532, an RTC 528, an (optional) encryption/decryption engine 522, an (optional) random number generator (RNG) 542, an (optional) arithmetic accelerator 544, and a (optional) compression/decompression engine 546, and a (optional) pattern matching circuit 524. Microprocessor 520 is omitted from SPU chip 2660 since the standard microprocessor 2650 performs the processing functions instead. In addition, SPU chip 2660 may include a flag 2671 and AND gate logic 2672 for the initialization purposes discussed above. x. In addition, SPU chip 2660 includes an enhanced switch 2663 that provides the same overall (bus enhanced) functionality performed by the switch 2658 in the FIG. 9A embodiment. y. Enhanced switch 2663 would perform the functions of a bus repeater, mediator and interpreter. For example, enhanced switch 2663 may act as a bus repeater that enables microprocessor 2652's memory accesses made over internal memory bus 2661 to be reflected to external memory bus 2664 and performed on main memory 2665. Enhanced switch 2663 may also act as a bus repeater similarly for internal I/O bus 2662 to external I/O bus 2665 in the event that microprocessor 2652 performs I/O operations distinctly from memory operations. Enhanced switch 2663 may also perform the function of a mediator for microprocessor control functions 2666 (e.g., non-maskable interrupt, reset) with respect to externally requested control functions 2667. Enhanced switch 2663 may also provide mediation for access to SPU-protected resources such as ROM 532, RAM 534, encrypt/decrypt engine 522 (if present), random number generator 542 (if present), arithmetic accelerator 544 (if present), pattern matching engine 524 (if present), and real-time clock 528 (if present). Enhanced switch 2663 may also act as an interpreter of control signals received from microprocessor 2652 indicating entry to, exit from, and control of SPU mode. z. Switch 2663 in this example recognizes a specific indication (e.g., an instruction fetch access to a designated address in the secure memory 532) as the equivalent to the 'enable `SPU` mode' instruction. Upon recognizing such an indication, it may isolate the CPU/SPU 2650 from external buses and interfaces 2664, 2665, and 2667 such that any external activity, such as DMA cycles, would be 'held' until the switch 2663 permits access again. After this, switch 2663 permits a single access to a specific location in secure memory 532 to complete. aa. The single instruction fetched from the designated location performs a control operation (a cache flush, for example), that can only be performed in microprocessor 2652's most privileged operating mode, and that has an effect visible to switch 2663. Switch 2663 awaits the occurrence of this event, and if it does not occur within the expected number of cycles, does not enter 'SPU' mode. bb. Occurrence of the control operation demonstrates that

	Claim Term/Phrase	Evidence Supporting MS Construction
		microprocessor 2652 is executing in its most privileged 'normal' mode and therefore can be trusted to execute successfully the 'enter `SPU` mode' sequence of instructions stored in secure memory 532. If microprocessor 2652 were not executing in its most privileged mode, there would be no assurance that those instructions would execute successfully. Because switch 2663 isolates microprocessor 2652 from external signals (e.g., interrupts) until 'SPU' mode is successfully initialized, the entry instructions can be guaranteed to complete successfully. cc. Following the initial instruction, switch 2663 can enter 'partial SPU mode,' in which a restricted area of ROM 532 and RAM 534 may be accessible. Subsequent instructions in secure memory 532 may then be executed by microprocessor 2652 to place it into a known state such that it can perform SPU functions--saving any previous state in the restricted area of RAM 534 that is accessible. After the known state is established, an instruction may be executed to deliver a further indication (e.g., a reference to another designated memory location) to switch 2663, which would enter 'SPU' mode. If this further indication is not received within the expected interval, switch 2663 will not enter 'SPU' mode. Once in 'SPU' mode, switch 2663 permits access to all of ROM 532, RAM 534, and other devices in SPU chip 2660. dd. The instructions executed during 'partial SPU' mode must be carefully selected to ensure that no similar combination of instructions and processor state could result in a control transfer out of the protected SPU code in ROM 532 or RAM 534. For example, internal debugging features of microprocessor 2652 must be disabled to ensure that a malicious program could not set up a breakpoint later within protected SPU code and receive control. Similarly, all address translation must be disabled or reinitialized to ensure that previously created MMU data structures would not permit SPU memory accesses to be compromised. The requirement that the instructions for 'partial SPU mode' run in the microprocessor 2652's most privileged mode is necessary to ensure that all its processor control functions can be effectively disabled. ee. The switch 2663 provides additional protection against tampering by ensuring that the expected control signals occur after an appropriate number of clock cycles. Because the 'partial SPU' initialization sequence is entirely deterministic, it is not feasible for malicious software to interfere with it and still retain the same timing characteristics, even if malicious software is running in microprocessor 2652's most privileged mode. ff. Once in 'SPU' mode, switch 2663 may respond to additional indications or signals generated by microprocessor 2652 (e.g., references to specific memory addresses) controlling features of SPU mode. These might include enabling access to external buses 2664 and 2665 so that SPU-protected code could reference external memory or devices. Any attempts by components outside CPU/SPU 2650 to perform operations

	Claim Term/Phrase	Evidence Supporting MS Construction
		(e.g., accesses to memory, interrupts, or other control functions) may be prevented by switch 2663 unless they had been explicitly enabled by instructions executed after 'SPU' mode is entered. To leave SPU mode and return to normal operation, the instructions executing in 'SPU' mode may provide a specific indication to switch 2663 (e.g., a transfer to a designated memory address). This indication may be recognized by switch 2663 as indicating a return to 'normal mode,' and it may again restrict access to ROM 532, RAM 534, and all other devices within SPU chip 2660, while re-enabling external buses and control lines 2664, 2665, and 2667. The instructions executed subsequently may restore the CPU state to that which was saved on entry to SPU mode, so that microprocessor 2652 may continue to perform functions in progress when the SPU was invoked. gg. In an alternate embodiment, the entry into SPU mode may be conditioned on an indication recognized by switch 2663, but the switch may then use a hardware mechanism (e.g., the processor's RESET signal) to reinitialize microprocessor 2562. In such an embodiment, switch 2663 may not implement partial SPU mode, but may instead enter SPU mode directly and ensure that the address from which instructions would be fetched by microprocessor 2652 (specific to microprocessor 2652's architecture) results in accesses to appropriate locations in the SPU memory 532. This could reduce the complexity of the SPU mode entry mechanisms in switch 2663, but could incur an additional processing cost from using a different reinitialization mechanism for microprocessor 2652. hh. SPU chip 2660 may be customized to operate in conjunction with a particular commercial microprocessor. In this example, the SPU may be customized to contain at least the specialized 'enter SPU mode' instruction sequences to reinitialize the processor's state and, to recognize special indications for SPU control operations. SPU chip 2660 may also be made electrically compatible with microprocessor 2652's external bus interfaces. This compatibility would permit CPU/SPU 2650 to be substituted for microprocessor 2652 without change either to software or hardware elsewhere in a computer system. ii. In other alternate embodiments, the functions described above for SPU chip 2600, microprocessor 2652, and internal buses 2661, 2662, and 2666 could all be combined within a single integrated circuit package, and/or on a single silicon die. This could reduce packaging complexity and/or simplify establishment of the hardware tamper-resistant barrier 502. jj. The hardware configuration of an example of electronic appliance 600 has been described above. The following section describes an example of the software architecture of electronic appliance 600 provided by the preferred embodiment, including the structure and operation of preferred embodiment 'Rights Operating System' ('ROS') 602."

	Claim Term/Phrase	Evidence Supporting MS Construction
		2. See '900 230:55 – 233:34 a. "Integrity of Software-Based PPE Security b. As discussed above in connection with FIG. 10, some applications may use a software-based protected processing environment 650 (such as a 'host event processing environment' (HPE) 655) providing a software-based tamper resistant barrier 674. Software-based tamper resistant barrier 674 may be created by software executing on a general-purpose CPU. Various software protection techniques may be used to construct and/or provide software-based tamper resistant barrier 674. c. The risks or threat of attacks described above in connection with PPE 650 apply to a software-based PPE. An important threat to be countered with respect to a software-based tamper resistant barrier 674 is an attack based on a distributable computer program that can defeat the tamper resistant barrier wherever the program is run. Since a software-based tamper resistant barrier 674 typically will not be as secure as a hardware-based tamper resistant barrier 502, it is useful to explore example steps and procedures a 'cracker' might use to "crack" a software'-based tamper resistant barrier. d. FIGS. 67A and 67B show example 'cracking' techniques a 'cracker' might use to attack software-based tamper resistant barrier 674. e. Referring to FIG. 67A, the software used to create tamper resistant barrier 674 may be distributed, for example, on a storage medium 3370 such as a floppy diskette or optical disk (or, this software could be distributed electronically over network 108 and stored locally in a computer memory). The software distribution medium 3370 provides software (code and data) for loading into a computing device such as a general purpose personal computer 3372, for example. Personal computer 3372 may include, for example, a random access memory 3374 and a hard disk 3376. f. In one example, the software distribution medium 3370 might include installation materials 3470 and operational materials 3472. The installation materials 3470 may be executed by computer 3372 to install the operational materials 3472 onto the computer's hard disk 3376. The computer 3372 may then execute the operational materials 3472 from its hard disk 3376 to provide software-based protected processing environment 650 and associated software-based tamper resistant barrier 672. g. In this example, one attack technique an attacker might use is to analyze software distribution medium 3370 (see FIG. 67B, block 3352). Such analysis can take many forms. h. Such analysis could be performed by a combination of one or more techniques. Such techniques include, but are not limited to, the following: i. An attacker can manually 'dump' and/or disassemble listings of the data from medium 3370. This analysis is represented in FIG. 67A by magnifying glass 3352A. j. An attacker can use cryptoanalytic and/or key search techniques to

	Claim Term/Phrase	Evidence Supporting MS Construction
		decrypt any encrypted data from medium 3370. k. An attacker can use automated or semi-automated disassembly tools to explore the functions of programs stored on medium 3370 by studying the operation and flow of the assembly language representation of the programs. This analysis is represented in FIG. 67A by block 3352B. l. An attacker can use software reverse-engineering tools to reconstruct high-level language representations of the programs on medium 3370, and study their functions. This analysis is represented in FIG. 67A by block 3352C, producing source code 3371. m. An attacker can use software reverse-engineering tools to create an equivalent program to the programs stored on medium 3370. As the equivalent program may be in a more convenient form, possibly in a higher-level language, it may be more amenable to analysis. This analysis is also represented in FIG. 67A by block 3352C, producing source code 3371. n. An attacker can use software debugging and/or simulation tools to follow and/or modify the dynamic execution of programs from medium 3370. This technique can be combined with any of the above static analysis techniques to study the program as it operates. This analysis is represented in FIG. 67A by block 3352B. o. An attacker can use hardware-based debugging and/or simulation tools (e.g., an in-circuit emulator, or ICE) to follow and/or modify the dynamic execution of programs from medium 3370. This technique may be more effective than the equivalent using software debugging and/or simulation tools because it has less potential effect on operation of the programs. This analysis is represented in FIG. 67A by block 3352B. p. Such analysis could provide clues and insights into the installation materials 3470, the operational materials 3472, or both. q. Another attack technique could focus on the operational materials 3472 in the form in which they are installed on personal computer 3372. For example, one form of analysis might involve analyzing the on-disk copy of the installed software and/or associated data files installed on computer hard disk 3376 (see FIG. 67B, block 3354). This analysis is represented in FIG. 67A as a magnifying glass 3354B. Because the installed operational materials 3472 can be executed by computer 3372, the analysis need not be limited to analyzing the static information stored on hard disk 3376, but could involve performing static and/or dynamic analysis of the executing software (see FIG. 67B, blocks 3356, 3358). Any of the techniques described above could be used to analyze the operational material software 3472 to yield source code or other more interpretable form 3373A and/or a memory image 3373B. The static and/or dynamic data within RAM 3374A could be similarly analyzed (see FIG. 67A, magnifying glass 3354A). r. The resulting source code 3373A and/or memory image 3373B could be carefully analyzed and reviewed (see magnifying glasses 3354D, 3354E) to obtain an understanding of both the static and dynamic structure and

	Claim Term/Phrase	Evidence Supporting MS Construction
		operation of operational materials 3272. Dynamic code analysis could involve, for example, tracing, single-stepping, data, or code break points of the executing software image, using analysis techniques such as described above. The executing software could be modified dynamically (for example, by patching) during normal operation to attempt to bypass its protection mechanisms and/or to learn more about how it operates (see FIG. 67B, block 3360, and the 'changes' inserted into FIG. 67A memory image 3373B). s. A further attack technique in this example might involve comparing installed operational material 3472 software and data files among several different PPE 650 instances to identify important data structures, such as cryptographic keys (see 'compare' block 3362A of FIG. 67A; and FIG. 67B, block 3362). The resulting list of differences 3362B could be carefully analyzed (see FIG. 67A's magnifying glass 3362C) to obtain important clues, using analysis techniques such as described above. t. A further attack technique might involve comparing the memory and/or disk images of installed operational material 3472 software and data files in a single instance of PPE 650, after performing various operations using the PPE. This could serve to identify important data structures, such as cryptographic keys (see 'compare' block 3362A of FIG. 67A; and FIG. 67B, block 3362). The resulting list of differences 3362B could be carefully analyzed (see FIG. 67A's magnifying glass 3362C) to obtain important clues, using analysis techniques such as described above. u. A further attack technique might involve analyzing the timing and/or order of modification to memory and/or disk images of installed operational material 3472 software and data files in a single instance of PPE 650, during the performance performing various operations using the PPE. This could serve to identify important data structures, such as cryptographic keys (see 'compare' block 3362A of FIG. 67A; and FIG. 67B, block 3362). The resulting list of differences 3362B could be carefully analyzed (see FIG. 67A's magnifying glass 3362C) to obtain important clues, using analysis techniques such as described above. v. A further attack technique might involve duplicating one installed operational material 3472 instance by copying the programs and data from one personal computer 3372B to another personal computer 3372C or emulator (see FIG. 67B, block 3364, and the 'copy' arrow 3364A in FIG. 67A). The duplicated PPE instance could be used in a variety of ways, such as, for example, to place an impostor PPE 650 instance on-line and/or to permit further dynamic analysis. w. A still additional avenue of attack might involve, for example, saving the state of a PPE 650 (see FIG. 67A, block 3366B)--for example, before the expenditure of credit--and restoring the state at a subsequent time (e.g., after a payment operation occurs) (see FIG. 67A, arrows 3366A, 3366C, and FIG. 67B, block 3366). The stored state information 3366B may also be analyzed (see FIG. 67A, magnifying glass 3354F. x. No software-only tamper resistant barrier 674 can be wholly effective

	Claim Term/Phrase	Evidence Supporting MS Construction
		against all of these threats. A sufficiently powerful dynamic analysis (such as one employing an in-circuit emulator) can lay bare all of the software-based PPE 650's secrets. Nonetheless, various techniques described below in connection with FIG. 69A and following make such an analysis extremely frustrating and time consuming--increasing the 'work factor' to a point where it may become commercially unfeasible to attempt to 'crack' a software-based tamper resistant barrier 674." 3. See '900 235:28 – 244:15 a. "Example Techniques for Forming Software-Based Tamper Resistant Barrier b. Various software protection techniques detailed above in connection with FIG. 10 may provide software-based tamper resistant barrier 674 within a software-only and/or hybrid software/hardware protected processing environment 650. The following is an elaboration on those above-described techniques. These software protection techniques may provide, for example, the following: c. An on-line registration process that results in the creation of a shared secret between the registry and the PPE 650 instance--used by the registry to create content and transactions that are meaningful only to that specific PPE instance. d. An installation program (that may be distinct from the PPE operational material software) that creates a customized installation of the PPE software unique to each PPE instance and/or associated electronic appliance 600. e. Camouflage protections that make it difficult to reverse engineer the PPE 650 operational materials during PPE operation. f. Integrity checks performed during PPE 650 operation (e.g., during on-line interactions with trusted servers) to detect compromise and minimize damage associated with any compromise. g. In general, the software-based tamper resistant barrier 674 may establish 'trust' primarily through uniqueness and complexity. In particular, uniqueness and customization complicate the ability of an attacker to: make multiple PPE instances with the same apparent identity; make it harder for an attacker to create a software program(s) that will defeat the tamper-resistant barrier 674 of multiple PPE instances; make it harder for the attacker to reverse engineer (e.g., based upon encryption so that normal debugging/emulation and other software testing tools can't easily provide access); and make it more difficult for an attacker to compare multiple PPE instances to determine differences between them. h. In addition, the overall software-based tamper resistant barrier 674 and associated PPE system is sufficiently complex so that it is difficult to tamper with a part of it without destroying other aspects of its functionality (i.e., a 'defense in depth'). Camouflaging techniques complicate an attacker's analysis through use of debugging/emulation or other software tools. For example, the PPE 650 may rewrite or overwrite

	Claim Term/Phrase	Evidence Supporting MS Construction
		memory locations immediately after using same to make their contents unavailable for scrutiny. Similarly, the PPE 650 operational software may use hardware and/or time dependent sequences to prevent emulation. Additionally, some of the PPE 650 environment code may be self-modifying. These and other techniques make it much harder to crack an individual PPE 650 instance, and more importantly--much harder to write a program that could be used to defeat security on multiple PPE instances. Because the legitimate owner/user of a particular PPE instance may be trying to attack the security of his own system, these techniques assume that individual instances may eventually be cracked and provide additional security and safeguards that prevent (or make it more difficult) for the attacker who has cracked one PPE instance to use that information successfully in cracking other PPE instances. Specifically, these security techniques make it unlikely that an attacker who has successfully cracked one or a small number of PPE instances can write a program capable of compromising the security of any arbitrary other PPE instance, for example. i. Example Installation Process j. Briefly, the preferred example software-based PPE 650 installation process provides the following security techniques: encrypted software distribution, installation customized on a unique instance and/or electronic appliance basis, encrypted on-disk form, installation tied to payment method, unique software and data layout, and identifiable copies. k. FIG. 69A shows one example technique for distributing the PPE 650 software. In this example, the PPE 650 software is distributed as two separate parts and/or media: the installation materials 3470, and the operational materials 3472. Installation materials 3470 may provide executable code and associated data structures for installing the operational materials 3472 onto a personal computer hard disk 3376, for example (see FIG. 67A). The operational materials 3472 may provide executable code and associated data structures for providing protected processing environment 650 and associated software-based tamper resistant barrier 674. l. In this example, installation materials 3470 and operational materials 3472 are each encrypted by a 'deliverable preparation' process 3474 to provide encrypted installation materials 3470E and encrypted operational materials 3472E (the encrypted portions are indicated in FIG. 69A, by cross-hatching). In this example, a small portion 3470C of the installation materials 3470 may be maintained in clear (unencrypted) form to provide an initial portion of the installation routine that may be executed without decryption. This plain text portion 3470C may, for example, provide an initial dialog, using an encrypted or other secure protocol with a trusted

	Claim Term/Phrase	Evidence Supporting MS Construction
		registry 3476 such as VDE administrator 200h for example. This makes the distributed installation materials 3470 and operational materials 3472 meaningless and unreadable to an attacker without additional information since the entire content (except for the initial dialog with the registry 3476) is unreadable. m. In this example, the 'deliverable preparation' process 3474 may encrypt the installation materials 3470 and operational materials 3472 using one or more secret keys known to the registry 3476. Multiple versions of these installation materials 3470 and operational materials 3472 may be distributed using different, secret keys so that compromise of one key exposes only a subset of the software distribution to unwanted disclosure. The only non-encrypted part of the software distribution in plaintext is that portion 3470C of installation materials 3470 used to establish initial contact with the registry 3476. n. The registry 3476 maintains a copy of the corresponding decryption keys within a key generation and cataloging structure 3478. It provides these keys on demand during the registration process (e.g., using a secure key exchange protocol, for example) to only legitimate users authorized to set up a new protected processing environment 650. o. FIGS. 69B-69C show example steps that may be performed by a installation routine 3470 to install a protected processing environment 650. In this example, upon coupling the installation materials 3470 to an electronic appliance 600 such as a personal computer 3372, the appliance begins executing the unencrypted installation materials portion 3470C. This plain text portion 3470C controls appliance 600 to contact registry 3476 and establish a registry dialog (FIG. 69B, block 3470(1)). The appliance 600 and the registry 3476 use a secure key exchange protocol to exchange installation keys so that the registry may deliver the appropriate installation key to the appliance (FIG. 69B, block 3470(2)). Using the provided installation key(s), the appliance 600 may decrypt and run additional portions of encrypted installation materials 3470E (FIG. 69B, block 3470(3) and following). Based on this additional installation program execution, appliance 600 may decrypt and install encrypted operational materials 3472E (FIG. 69B, block 3470(4)). p. Rather than simply installing the operational materials 3472, in one example, installation materials 3470 makes the installation different for each PPE 650 instance. For example, the installation materials 3470 may customize the installation by: uniquely embedding important data into the installed software, uniquely encrypting the installed software, uniquely making random changes to the installed software, uniquely mating the installed software with a particular electronic appliance 600, providing a unique static and/or dynamic layout or other structure. q. Randomly Embedded Cryptographic Keys r. Installation routine 3470 may, for example, modify the operational

	Claim Term/Phrase	Evidence Supporting MS Construction
		materials 3472 to customize embedded locations where critical data such as cryptographic keys are stored. These keys may be embedded into the text of the operational materials 3472 at locations that vary with each installation. In this example, the registry 3476 may choose, on a random or pseudo-random basis, at least some of the operational material 3472 locations in which a particular installation routine 3470 may embed cryptographic keys or other critical data (see FIG. 69B, block 3470(5)). s. The installation process for the operational software may involve decrypting its distribution (which may be the same for all end users) and modifying it to encode the specific locations where its critical data (e.g., cryptographic keys) are stored. These keys may be embedded within the text of the program at locations that vary with every installation. The distribution of unique information into the operational software 3472 can be based on a secret key known to the registry 3476. This key may be communicated by the registry 3476 during the registration dialog using a secure key exchange. The key is shared between the registry 3476 and the PPE 650 instance, and can serve both to organize the installed PPE software, and as the basis of subsequent integrity checks. t. As shown in FIG. 69D, the operational materials 3472 may include embedded locations 3480(a), 3480(b), 3480(c), 3480(d), 3480(e), ... reserved for storing (embedding) critical information such as cryptographic keys. Each of these locations 3480 may initially store a random number string. In one example, the registry 3476 or installation routine 3470 performs a random operation 3482 to randomly select which subset of these locations 3480 is to be used by a particular instance for storing critical data. This selection list 3484 is applied as an input to an operation materials preparation step 3474a (part of the deliverable preparation operation 3474 shown in FIG. 69A). The operation materials preparation step 3474a also accepts, as an input, cryptographic keys from a secure key store 3486. In this example, the operation materials preparation step 3474a embeds the cryptographic keys provided by key store 3486 into the selected locations 3484 of operation materials 3472. u. In accordance with one example, the random operation 3482 selects a subset that is much less than all of the possible locations 3480--and the locations 3480 not used for storing cryptographic keys store random data instead. An attacker attempting to analyze installed operational materials 3472 won't be able to tell the difference between real cryptographic keys and random number strings inserted into a place where cryptographic keys might be stored. v. In this example, the random location selection 3484 (which is unique for each installation) may itself be encrypted by block 3488 based on an installation-unique key provided by key generation block 3490 for example. The encryption key may be securely maintained at registry 3476 so that the registry may later notify the installation materials 3470 of this key--allowing the installation materials to decrypt the resulting encrypted key location block 3492 and recover listing 3484 of the subset

	Claim Term/Phrase	Evidence Supporting MS Construction
		of locations 3480 used for embedding cryptographic keys. w. Embedded Customized Random Changes x. Referring once again to FIG. 69B, the installed operational materials 3472 may be further customized for each instance by making random changes to reserved, unused portions of the operational materials (FIG. 69B, block 3470(6)). An example of this is shown in FIG. 69E. In this example, the operational materials 3472 include unused, embedded random data or code portions 3494. Another technique with similar effect is shown in FIG. 69F. In this example, false code sections 3496 are included within reserved areas of the operational materials 3472. These false code sections 3496 add complexity, and may also be used as a electronic 'fingerprint' to help trace copies. Because the false code sections 3496 are executable program code that are never executed (or if executed perform no actual functions other than confounding analysis by, for example, creating, modifying and/or destroying data that has no impact on the operation of PPE 650 but may appear to have such an impact), they can be used to confound analysis because they may be difficult for an attacker to distinguish from true code sections. In addition other false code may have the effect of disabling the execution of PPE 650 if executed. Correspondence Between Installed Software and Appliance 'Signature'. Another technique that may be used during the installation routine 3470 is to customize the operational materials 3472 by embedding a 'machine signature' into the operational materials to establish a correspondence between the installed software on a particular electronic appliance 600 (FIG. 69C, block 3470(7)). This technique prevents a software-based PPE 650 from being transferred from one electronic appliance 600 to another (except through the use of the appropriate secure, verified backup mechanism). y. For electronic appliances 600 where it is feasible to do so, the installation procedure 3470 may determine unique information about the electronic appliance 600 (e.g., a 'signature' SIG in the sense of a unique value--not necessarily a 'digital signature' in the cryptographic sense). Installation routine 3470 embeds the electronic appliance 'signature' SIG in the installed operational materials 3472. Upon initialization, the operational materials 3472 validate the embedded signature value against the actual electronic appliance 600 signature SIG, and may refuse to start if the comparison fails. z. Depending on the configuration of electronic appliance 600, the machine signature may consist, for example, of some combination of a hash of the ROM BIOS 658' (see FIG. 69G), a hash of a disk defect map 3497a, the Ethernet (or other) network adapter 666 address, information written into an unused disk sector, information stored in a non-volatile CMOS RAM(such as used for hardware configuration data), information stored in non-volatile ('flash') memory (such as used for system or peripheral component 'BIOS' programs) and/or hidden unique information placed into the root directory 3497b of the fixed disk drive 668.

	Claim Term/Phrase	Evidence Supporting MS Construction
		aa. FIG. 69G shows an example of some of these appliance-specific signatures. bb. In this example, machine signature information need not be particularly large. Security is provided by hiding the machine signature rather than on any other cryptographic strength, because there is no more secure mechanism for key storage to protect it. Thus, it is satisfactory for the signature to be just large enough (e.g., two bytes) that it is unlikely to be duplicated by chance. cc. For some electronic appliances 600 where it can be determined that the technique is safe, an otherwise unused section of the non-volatile CMOS RAM 656a may be used to store a signature 3497d. Signature 3497d is verified against the PPE 650's internal state whenever the PPE is initialized. Signature 3497d may also be updated whenever a significant change is made to the secure database 610. If the CMOS RAM signature 3497d does not match the database value, PPE 650 may take this mismatch as an indication that a previous instance of the secure database 610 and/or PPE 650 software has been restored, and appropriate action can be taken. This mechanism thus ensures that even a bit-for-bit copy of the system's fixed disk 668 or other storage medium cannot be saved and reloaded to restore an earlier PPE 650 state. This particular technique depends upon there being an unused location available within CMOS RAM 656a, and may also require the CMOS RAM checksum algorithm to be known. An incorrect implementation could cause a subsequent reboot of electronic appliance 600 to fail because of a bad CMOS checksum, or worse, could alter some critical configuration parameter within CMOS RAM 656a so that electronic appliance 600 could not be recovered. Thus, care must be taken before modifying the contents of CMOS RAM 656a. dd. A still alternate technique may involve marking otherwise 'good' disk sectors 3497c defective and using the sector(s) to store machine signatures and/or encryption keys. This technique ensures that a logical bit-for-bit copy of the media does not result in a usable PPE 650 instance, and also provides relatively inaccessible and non-volatile storage for the information. Because a relatively large amount of storage space can be reserved using this technique, there is enough storage for a cryptographically strong value. ee. Some of the 'machine signature' techniques discussed above may be problematic in some electronic appliances 600 because it may be difficult to locate appropriate appliance-unique information. For example, although in a personal computer a ROM BIOS 658' is always available, the ROM BIOS information by itself may be insufficient because it is likely to be identical for a batch of electronic appliances 600 purchased together. Identifying a network adapter 666 and determining its address is potentially difficult due to the wide variety of adapters; additionally, an electronic appliance's network address may change (although this occurrence may be infrequent). Inserting random signature values into unused bytes within the fixed disk root directory 3497b and/or partition

	Claim Term/Phrase	Evidence Supporting MS Construction
		records may trigger some virus-checking programs, and the data may be modified by defragmentation or other disk manipulation programs. Where supported, a truly unused disk sector 3497c (e.g., one that is marked 'bad' even though it may still viably store information) may be used to store the machine signature. Even so, normal maintenance, upgrades or other failure recovery procedures may disrupt a particular machine association. Since the VDE administrator 200h participates in restoring a PPE 650 based on an encrypted backup image (as described above for example in connection with FIGS. 39-40), the VDE administrator may establish new associations at this point to maintain correspondence between a particular PPE 650 installation and a particular electronic appliance 600. ff. Tie Installation to Payment Method gg. A still additional example technique for providing additional security is to tie a particular PPE 650 installation at registration time to a particular payment method (see FIG. 69C, block 3470(8)). The registration process at installation time may thus serve to tie the PPE 650 installation to some payment method associated with the user, and to store the payment association information both within the PPE 650 instance and at the registry 3476. This technique assures that the actions of a particular PPE 650 instance are accountable to the assigned user with at least the reliability of whatever payment/credit verification technique is employed. hh. Install Operational Materials in Encrypted Form ii. Operational materials 3472 may first be customized as described above for the particular instance and/or appliance 600, then (at least mostly) encrypted for installation into the appliance such as by storage onto disk 668 (see FIG. 69C, block 3470(9)). Different installations may use different sets of decryption keys to decrypt the information once installed. Different parts of operational materials 3472 may be encrypted with different cryptographic keys to further complicate the analysis. This encryption makes analysis of the on disk form of the operational materials 3472 more difficult or infeasible. jj. The beginning of the resulting stored executable file may contain a small decryption program ('decryptor') that decrypts the remainder of the operational materials 3472 as they are loaded into memory. Confounding algorithms (as described below) may be used in this decryptor to make static recovery of the cryptographic keys difficult. Although the decryptor is necessarily in unencrypted form in an all-software installation without hardware support, the use of confounding algorithms to develop the associated cryptographic keys effectively requires a memory image to be captured after the program has been decrypted. Where supported (as described above), an unused and inaccessible disk sector 3497c may be used to store the decryption keys, and the operational materials 3472 may possess only the address for that particular sector. Embedding this address further complicates analysis. kk. Customized Layout

	Claim Term/Phrase	Evidence Supporting MS Construction
		ll. The installation materials 3470 may store the encrypted operational materials 3472 onto the fixed disk 668 using a customized storage layout (FIG. 69C, block 3470(10)). FIG. 69F, 69H, 69I and 69J shows example customized software and data layouts. In these examples, each installed instance of operational materials 3472 is different in both executable form and in data layout. These modifications make each PPE 650 instance require separate analysis in order to determine the storage locations of its critical data such as cryptographic keys. This technique is an effective counter to creation of programs that can undo the protections of an arbitrary PPE 650 instance. mm. Instruction sequences within the operational materials 3472 may be modified by the installation routine to change the execution flow of the executable operational materials 3472 and to alter the locations at which the software expects to locate critical data. The alterations in program flow may include customization of time-consuming confounding algorithms. The locations of the modifiable instruction sequences may be embedded within operational materials 3470, and may therefore be not directly available from an examination of the installation and/or operational materials. nn. FIG. 69H shows one example operational materials 3472 executable code segment provided distinct processes 3498a, 3498b, 3498c, 3498d, 3498e. In this particular example, segment 3498a is executed first and segment 3498e is executed last, but the processes 3498b, 3498c and 3498d may be performed in any order (i.e., they are sequence independent processes). The installation materials 3470 may take advantage of this sequence independence by storing and/or executing them in different and/or depending upon the particular PPE instance 650. FIG. 69I, for example, shows a first static layout order, and FIG. 69J shows a second, different static layout order. Data elements associated with the executables may similarly be stored in different orders (as shown in FIGS. 69I, 69J) depending upon the particular installation. oo. Dynamic Protection Mechanisms pp. In addition to the more static protection mechanisms described above, dynamic protection mechanisms may be employed to complicate both static and dynamic analysis of the executable (executing) operational materials 3472. Such techniques include, for example: qq. implementation complexity, immediate overwriting, hardware dependent sequences, timing dependencies, confounding algorithms, random modifications, dynamic load module decryption, rr. on-line integrity checks, time integrity checks, machine association integrity checks, dynamic storage integrity checks, and hidden secret storage volatile secret storage internal consistency checks. ss. FIGS. 69K-69L show an example execution of operational materials 3472 that may employ some or all of these various dynamic protection mechanisms. tt. Upon starting execution (FIG. 69K, block 3550), the installed operational

	Claim Term/Phrase	Evidence Supporting MS Construction
		materials 3472 may run initialization code as described above that is used to decrypt the stored encrypted operational materials on an 'as needed' basis (FIG. 69K, block 3552). This initialization code may also check the current value of the real-time clock (FIG. 69K, block 3554). uu. Real Time Check/Validation vv. Operational materials 3472 may perform this time check, for example, to guard against replay attacks and to ensure that the electronic appliance 600's time is in reasonable agreement with that of the VDE administrator 200h or other trusted node. ww. FIG. 69M shows an example sequence of steps that may be performed by the 'check time' block 3554. In this example, PPE 650 uses secure communications (e.g. a cryptographic protocol) to obtain the current real time from a trusted server (FIG. 69M, block 3554a). PPE 650 may next ask the user if he or she wishes to reset the electronic appliance real-time clock 528 (which may, for example, be the real-time clock module within a personal computer or the like) so it is synchronized with the trusted server's time clock. xx. If the user responds affirmatively, PPE 650 may reset the time clock to agree with the real-time provided by the trusted server ('yes' exit to decision block 3554b, FIG. 69M, block 3554c). If the user responds that he or she does not want the real-time clock reset ('no' exit to decision block 3554b), then PPE 650 may calculate a delta value of the difference between the server's real-time clock and the electronic appliance's real-time clock 528 (FIG. 69M, block 3554d). In either case, PPE 650 may store the current time Tcurrent into a non-volatile storage location Tstore indicating the current real-time (FIG. 69M, block 3554e). yy. Referring again to FIG. 69K, PPE 650 can disable itself if there is too much (or the wrong type) of a difference between the trusted server's time and the electronic appliance's clock--since such differences can indicate replay attacks, the possibility that the PPE 650 has been restored based on a previous state, etc. For example, if desired, PPE 650 can generate a time check fail exception if the electronic appliance's real-time clock 528 disagrees with the trusted server's real-time by more than a certain amount of acceptable drift (FIG. 69K, 'yes' exit to decision block 3556). In the event of such an exception, PPE 650 may disable itself (FIG. 69K, block 3558) and require a dialog between the user and registry 3476 (or other authority)--providing additional protection against replay attacks and also detecting clock failures that could lead to incorrect operation or incorrect charges. zz. Dynamic Code Decryption and Data OverWriting aaa. Operational materials 3472 may then decrypt the next program segment dynamically (FIG. 69K, block 3460. The code may be decrypted dynamically when it is needed, then re-encrypted or overwritten and discarded when not in use. This mechanism increases the tamper-resistance of the executable code--thus providing additional tamper resistance for PPE operations. As mentioned above, different decryption

	Claim Term/Phrase	Evidence Supporting MS Construction
		keys may be required to decode different code portions, and the decryption keys can be installation-specific so that an attacker who successfully comprises the decryption key of one instance cannot use that information to compromise any other instance's decryption key(s). bbb. Once a portion of the operational materials 3472 has been decrypted (FIG. 69K, block 3560), that portion may immediately overwrite all initialization code in memory since it is no longer required (FIG. 69K, block 3562). The executing operational materials 3472 may similarly overwrite all unwrapped cryptographic keys once they are no longer needed, and may also overwrite expanded key information developed by initializing the cryptographic algorithms once no longer needed. These techniques minimize the amount of time during which usable key information is available for exposure in a memory snapshot--complicating all but the most dynamic of analysis efforts. Because all keys in permanent storage are either encrypted or otherwise camouflaged, no such treatment is required for I/O buffers. ccc. Dynamic Check of Association Between Appliance and PPE Instance ddd. The executing operational materials 3472 may next compare an embedded electronic appliance signature SIG' against the electronic appliance signature SIG stored in the electronic appliance itself (FIG. 69K, decision block 3564). As discussed above, this technique may be used to help prevent operational materials 3472 from operating on any electronic appliance 600 other than the one it was initially installed on. PPE 650 may disable operation if this machine signature check fails ('no' exit to decision block 3564, FIG. 69K; disable block 3566). eee. Self-Modifying and/or Hardware-Dependent Code Sequences fff. Executing operational materials 3472 may also employ self-modifying code sequences that cannot easily be emulated with a software debugger or single-stepping program (FIG. 69K, block 3568). These sequences may, for example, be dependent on specific models of electronic appliances 600, and may be patched into the operational materials 3472 as appropriate to installation materials 3470 based on tests performed during the installation process. Such hardware-dependent sequences may be used to ensure that critical algorithms yield different results when executed on the proper hardware as opposed to when executed on different hardware or under software control such as in a debugger or emulator. To prevent such hardware-dependent sequences from being readily recognizable from a static examination of the code, the sequences may be constructed at run time and then invoked so that they can be identified only by analysis of the instruction sequences actually executed. ggg. Dynamic Timing Checks hhh. Executing operational materials 3472 may also make dynamic timing checks on various code sequences, and refuse to operate if they do not execute within the expected interval (FIG. 69K, block 3570, decision block 3572, 'disable' block 3574). An incorrect execution time suggests that the operational materials 3472 are being externally manipulated

	Claim Term/Phrase	Evidence Supporting MS Construction
		and/or analyzed or traced in some manner (e.g., by a software emulator). This technique thus provides additional protection against dynamic analysis and/or modification. iii. The expected execution intervals associated with certain code sequences may be calculated during the installation procedure. Resulting test values may be embedded into the operational materials 3472. These timing tests may be integrated with time integrity tests and dynamic integrity checks to make it more difficult to bypass them simply by patching out the timing check. Care should be taken to eliminate false alarms due to concurrent system activity (e.g., other tasks and/or windows)." ('900 235:28 - 244:15) 4. See also '900 Figs. 69A-N
30.	912.8: "identifying at least one aspect of an execution space required for use and/or execution of the load module"	**Intrinsic:** 1. "For each site, the manufacturer generates a site ID 2821 and list of site characteristics 2822." ('193 209:55-57)

Appendix 1 to Exhibit D: Source Abbreviations

Intrinsic Evidence:

Abbreviated Reference	Full Citation or Title
'193	U.S. Patent No. 6,253,193
'683	U.S. Patent No. 6,185,683
'721	U.S. Patent No. 6,157,721
'861	U.S. Patent No. 5,920,861
'891	U.S. Patent No. 5,982,891
'900	U.S. Patent No. 5,892,900
'912	U.S. Patent No. 5,917,912
'712	U.S. Patent Application Serial No. 08/699,712
'107	U.S. Patent Application Serial No. 08/388,107

Extrinsic Evidence:

Abbreviated Reference	Full Citation or Title
Bishop	M. Bishop, Computer Security, Art & Science, (2003).
Booth	C. J. Booth, ed. The New IEEE Standard Dictionary of Electrical and Electronics Terms, 5th edition, (1993).
Davies	D.W. Davies and W.L. Price, Security for Computer Networks, (1984) MSI083423-MIS083443.
Denning	D. Denning, Cryptography and Data Security, (1983), MSI085569.
Dictionary of Computing	Dictionary of Computing, 3rd edition, Oxford University Press, (1990).
IBM	G. McDaniel, ed., IBM Dictionary of Computing, (1994).
Laplante	P. A. Laplante, ed., Dictionary of Computer Science, Engineering, and Technology (2001).
Longley	D. Longley, et al., Information Security: Dictionary of Concepts, Standards and Terms, (1992).
Neumann	P.G. Neumann, Computer Related Risks, (1995).
Pfleeger	C. P. Pfleeger, Security in Computing, (1989).
Que	C. Weisert, Que's Computer Programmer's Dictionary, (1993).
Russell	D. Russell and G.T. Gangemi, Computer Security Basics, (1991).
Webster's	D. Spencer, Webster's New World Dictionary of Computer Terms, 4th edition (1992).